Red Hat Summit

Connect

Kosteneffizienz in der GenAI

Wie Small Language Models die Leistung von LLMs übertreffen können

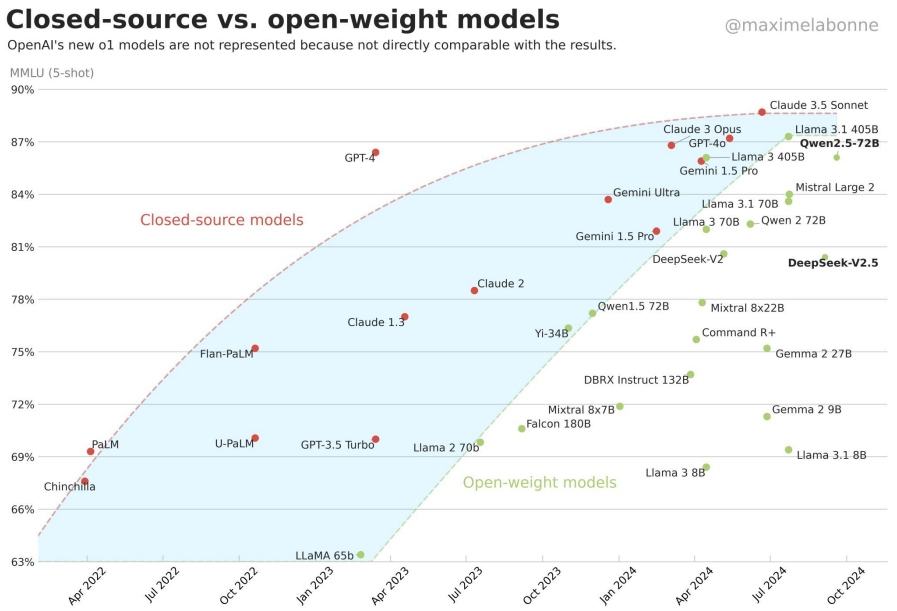
19.11.2025

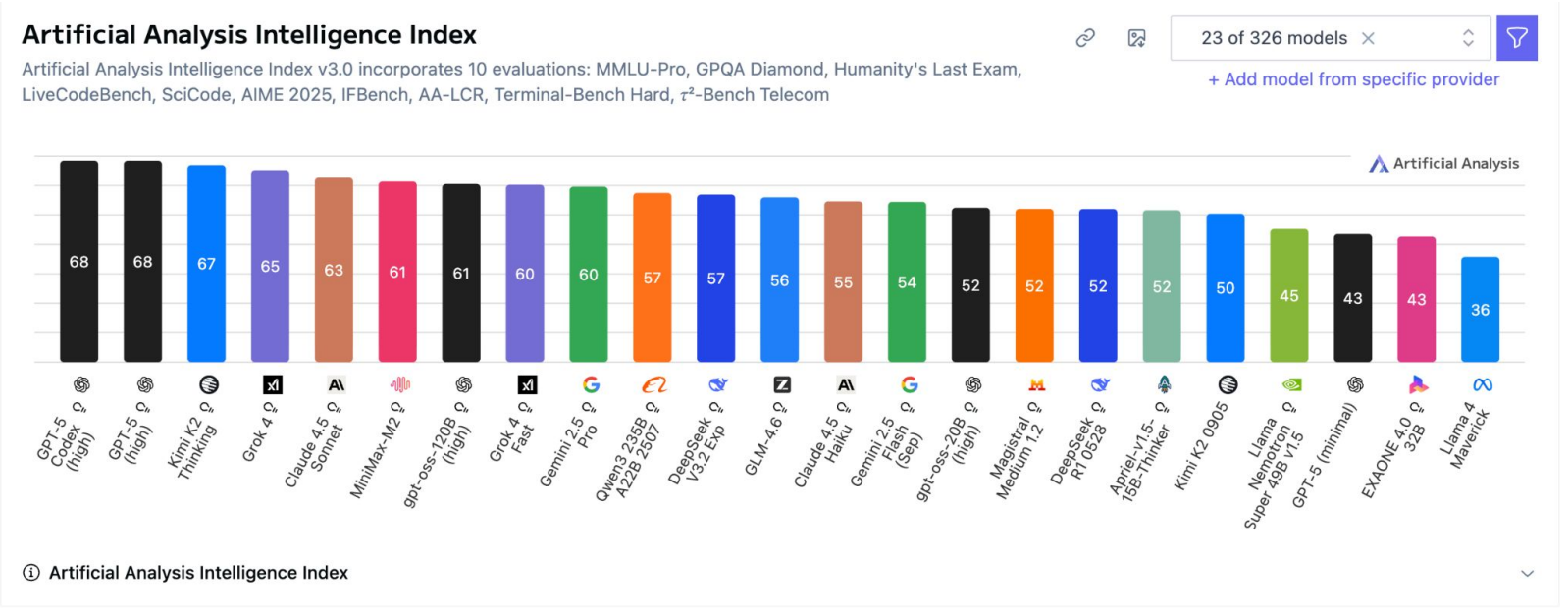
Darmstadt

Stefan Bergstein
Senior Principal Chief Architect

Steffen Röcker
Senior Account Solution Architect

Open Source Modelle auf dem Vormarsch

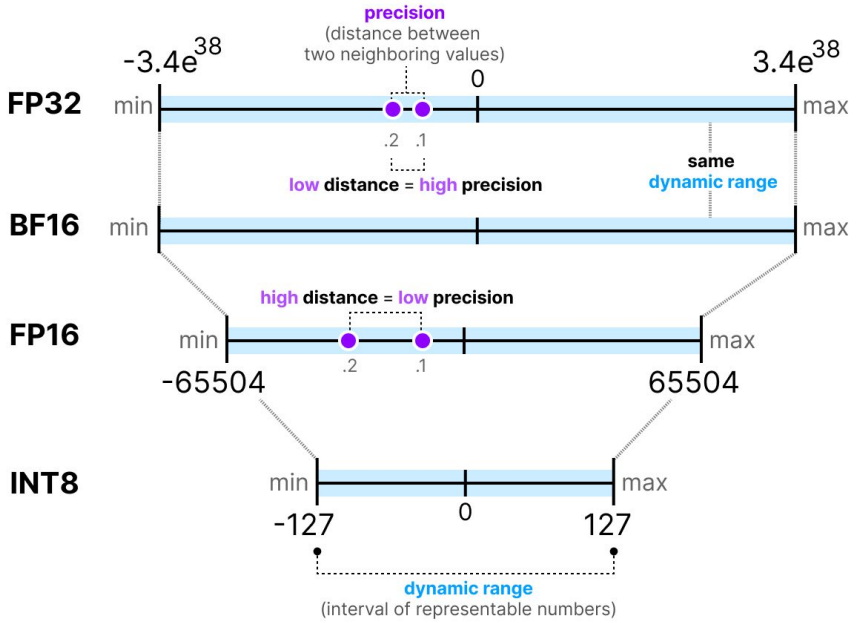




“Liebling ich habe das Sprachmodell geschrumpft”

Komprimierung a la MP3

- **Quantization:** Komprimiert Sprachmodelle indem die numerische Präzision verkleinert wird.
- Fließkommazahlen mit hoher Präzession (FP32) werden in platzsparende Formate (FP16, INT8, INT4) umgewandelt.
- **Pruning:** Schneidet unwichtige Teile weg
- **Reduziert den Speicher Footprint,** was Modelle einfacher zu deployen macht.



Was sind kleine Sprachmodelle?

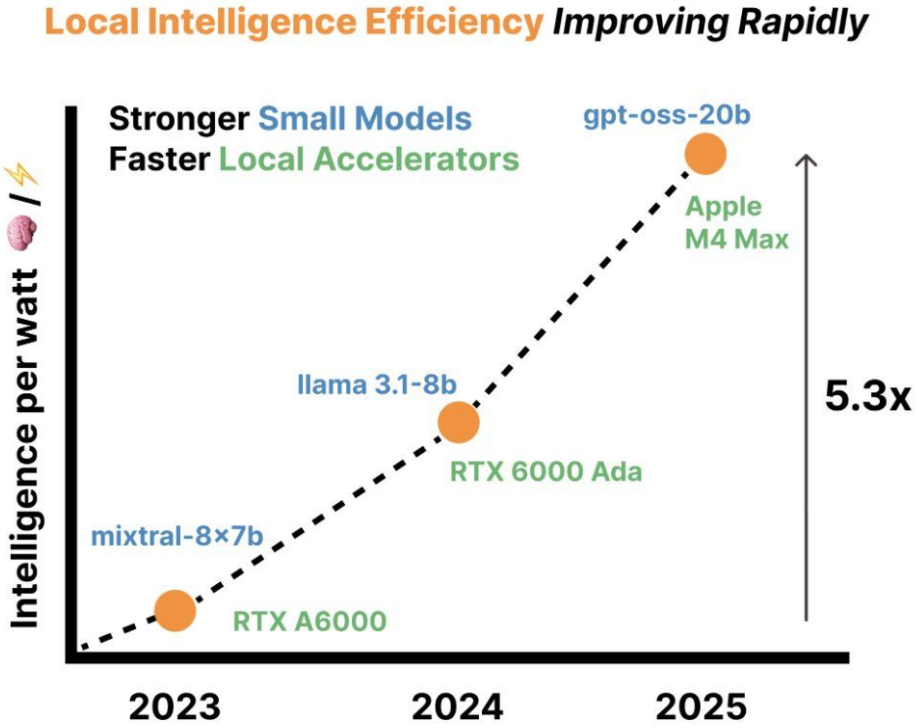
Small language models (SLMs):
Kleine Parameteranzahl von ein paar Millionen bis wenige Milliarden (<20B)

Laufen auf herkömmlicher Hardware, Edge Devices, inklusive Laptops und Handys
Vorteile: (Kosten)Effizient, Privat, Geringe Latenz, Anpassbar

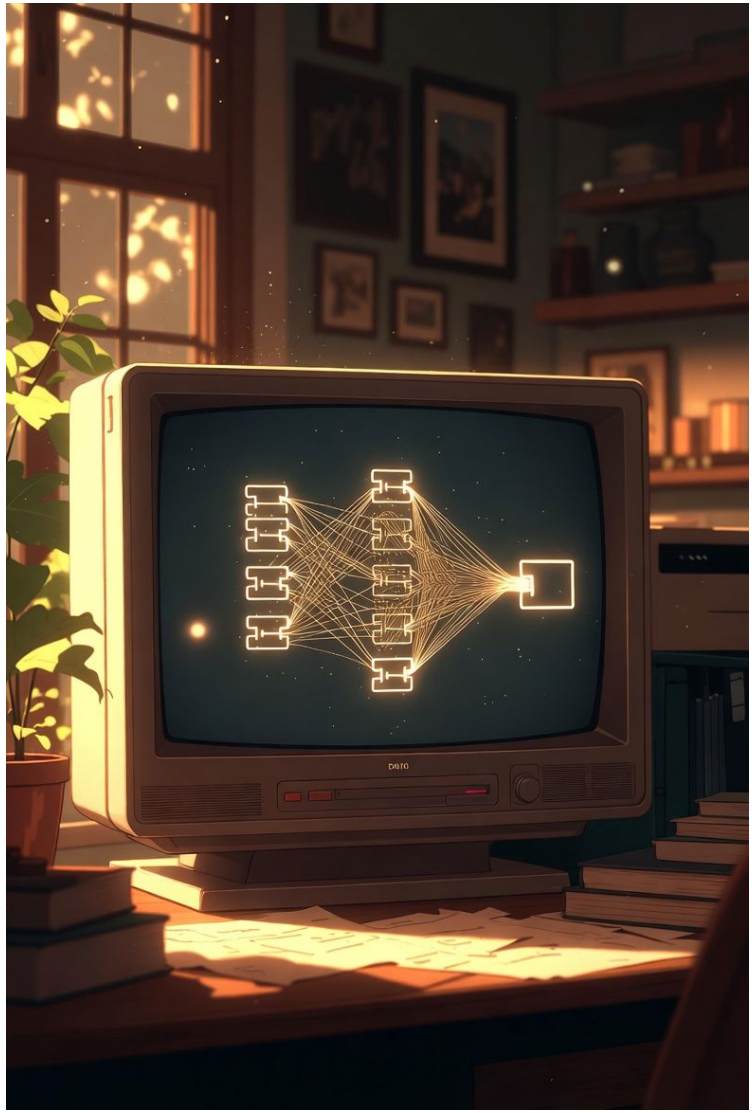
Bekannte SLMs:

- IBM Granite
- Qwen 2.5/3
- Google Phi-4 mini
- Google Gemma
- Meta Llama 3.2
- HF SmolLM
- TinyLama
- ModernBERT*
- ...

Und eine ganze Menge an spezialisierten SLMs



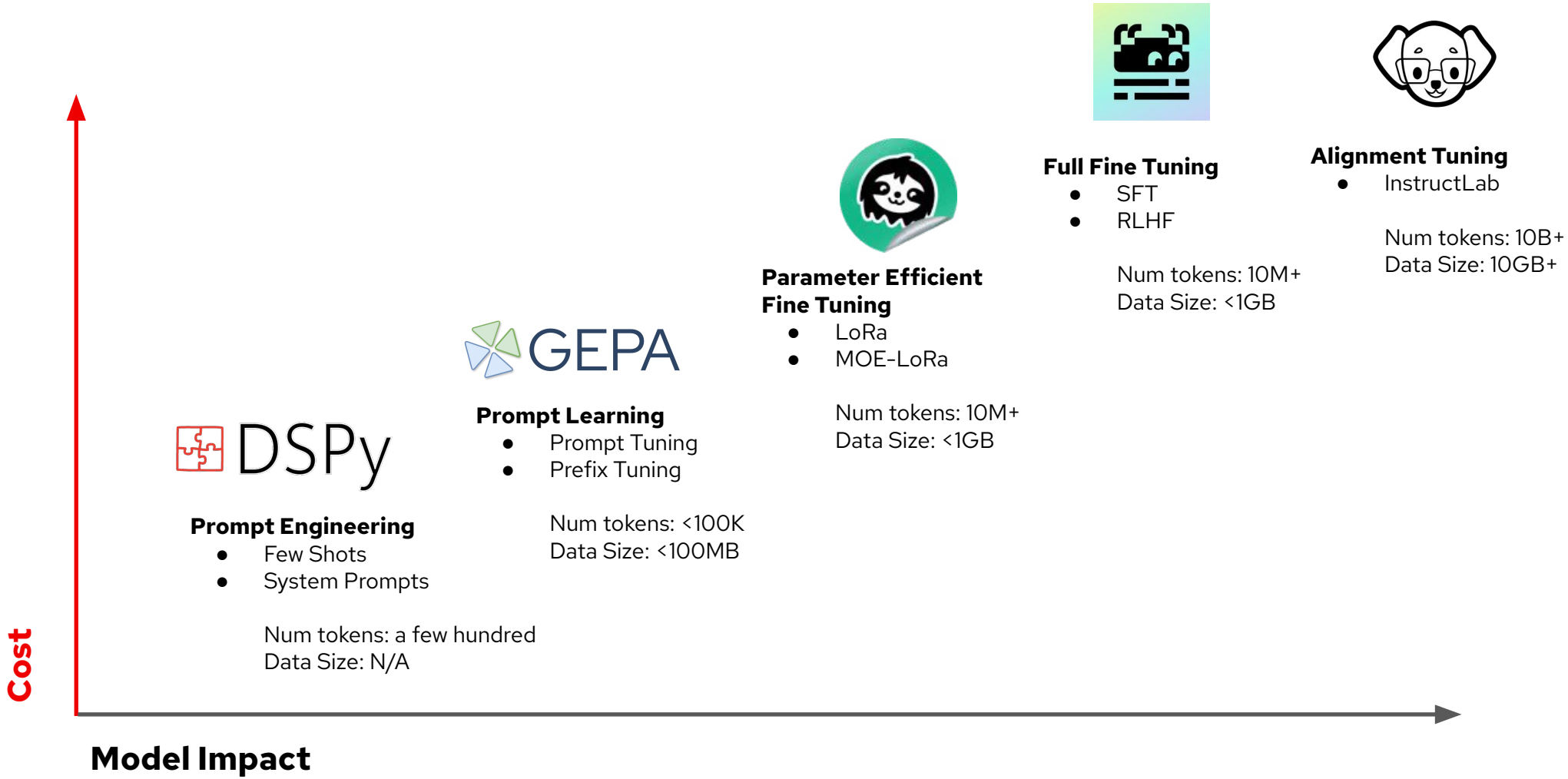
SLMs - Schnell und kostengünstig



Gut geeignet für:

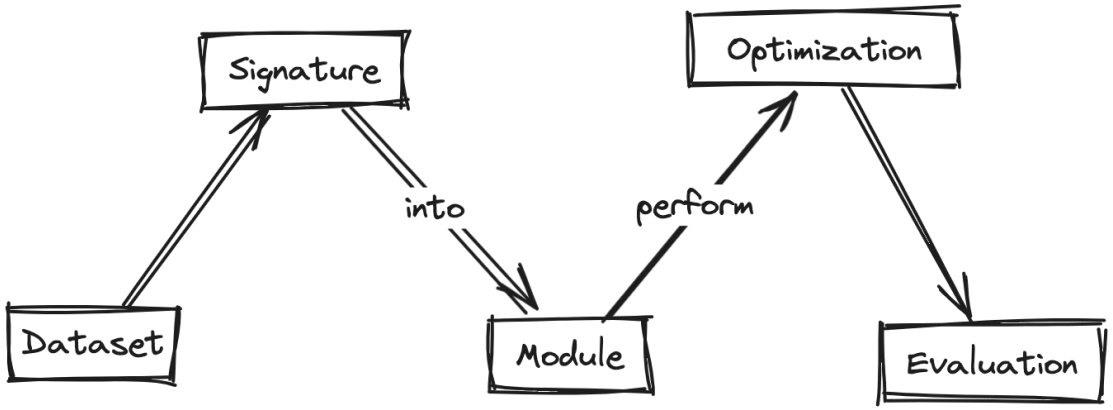
- Klassifizierung
  - Sentiment
  - Guardrails
  - Masking (PII und co)
- Zusammenfassung
- RAG (**Retrieval** Augmented Generation)
- OCR (Docling und co)
- Strukturierte Extraktion von Daten
- Agentic AI

Wie kann man SLMs spezialisieren?



Prompt Engineering

- Prompt Engineering ist eine Kunst für sich, lässt sich aber automatisieren
- Tools wie DSPy helfen dabei mit einem programmatischen Ansatz
- Prompts für kleine SLMs lassen sich durch mächtigere LLMs datenbasiert und iterativ verfeinern



`Classify if a resume is a good fit for a job description.`

The user wants you to act as a resume analysis assistant. Your task is to compare a given resume against a job description and determine if the candidate is a "Fit" or "No Fit".

Here's a breakdown of the process and key considerations:

Output Format:

- You must output exactly one of: "Fit" or "No Fit".
- You must provide a `classification_explanation` that justifies your decision.

Decision Criteria and Strategy:

1. **Prioritize Job-Related Evidence:** Focus strictly on skills, experience, and qualifications explicitly mentioned in the job description and directly evidenced in the resume.
2. **Be Strict:** Do not infer skills or experience. If a requirement is not clearly present in the resume, assume it is missing.
3. **Minimum Requirements are Crucial:** Pay close attention to "Minimum Requirements" or similar sections in the job description. If the resume does not meet these, it's a "No Fit."
4. **Years of Experience:** Carefully match the years of experience required for specific skills or roles. If the resume shows less experience than required, it's a "No Fit."
5. **Specific Technologies/Tools:** Look for exact matches for programming languages, software, platforms, methodologies (e.g., Agile, Scrum), and databases. General experience in a broad category (e.g., "software development") is not sufficient if specific technologies (e.g., "PERN stack," "Salesforce," "C++") are explicitly required.
6. **Domain-Specific Experience:** If the job description specifies experience in a particular industry (e.g., "Financial Services Industry," "Banking/Financial industry," "multifamily real estate," "energy systems"), the resume must clearly demonstrate this.
7. **Role-Specific Experience:** Ensure the candidate's experience aligns with the *type* of role described (e.g., "Business Analyst," "Software Data Engineer," "Multifamily Accountant," "Energy Systems Electrical Engineer"). A resume focused on administrative tasks will not fit a technical role, and vice-versa.
8. **Education/Certifications:** If specific degrees or certifications are required or strongly preferred, check for their presence.
9. **Soft Skills:** While important, soft skills (e.g., "excellent communication skills," "self-starter") should be secondary to technical and experience requirements unless explicitly highlighted as critical minimums.
10. **Avoid Generalizations:** Do not assume a candidate with broad IT experience automatically possesses specialized skills like "business process reengineering" or "Salesforce declarative and programmatic customization" unless explicitly stated.

Feedback Incorporation:

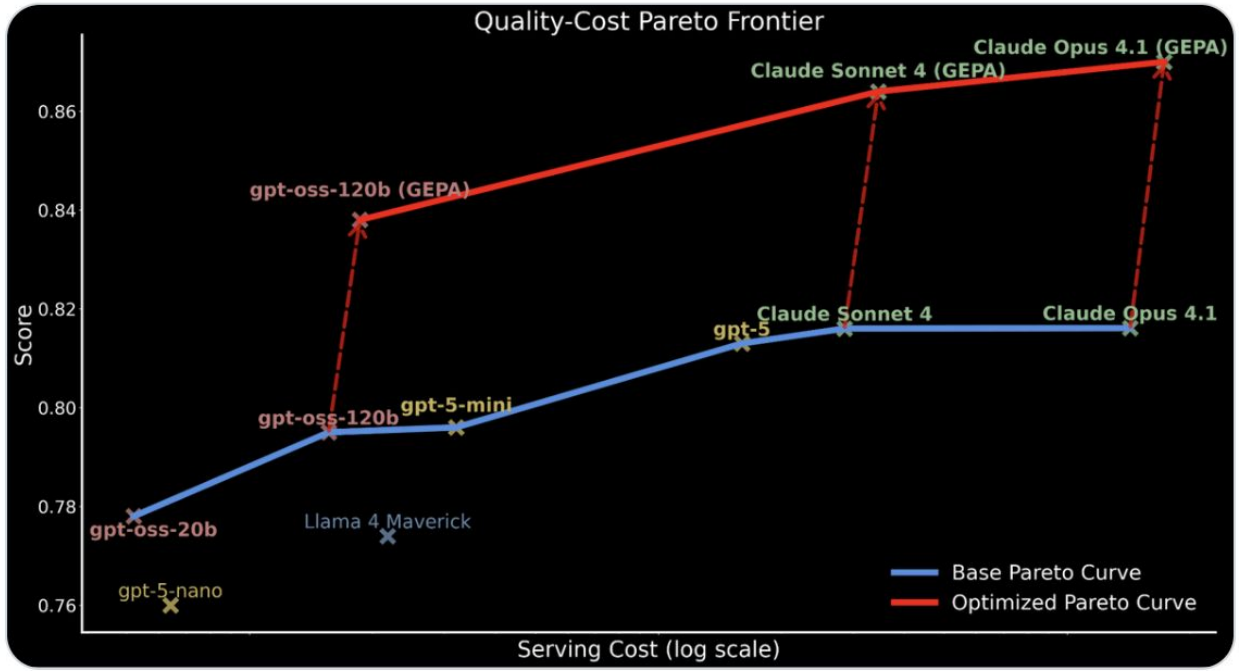
- **Example 5, 7, 12, 16, 22, 26, 28:** The model incorrectly classified these as "No Fit" when they were "Good Fit." This indicates that the model was too strict in its interpretation or missed subtle connections. For future tasks, ensure that if a resume *does* contain evidence for the core requirements, even if not perfectly aligned with every single keyword, it should lean towards "Fit" if the primary skills and experience are present. For instance, if "data analysis" is required, and the resume shows multiple roles with data analysis responsibilities, it should be considered a fit, even if specific tools like "SAS" are not explicitly mentioned but other analytical tools are. The model should be able to infer a fit if the *essence* of the required experience is there, especially for roles like Business Analyst where diverse backgrounds can be relevant.
- **Example 6, 10:** The model incorrectly classified these as "No Fit" when they were "Good Fit." This suggests the model might be overly focused on exact keyword matches for programming languages or specific tools, rather than recognizing equivalent or transferable skills and experience. For instance, if a job requires C++ and Python, and the resume shows strong Python experience and some C++ (even if not 5+ years *recent* C++), and the overall profile is senior software engineering, it might be a "Fit." The model needs to be more flexible in evaluating programming language experience, especially when multiple languages are listed as preferred or required, and the candidate demonstrates strong proficiency in a subset of them.
- **Example 25:** The model incorrectly classified this as "No Fit" when it was "Good Fit." This highlights the need to recognize when a candidate's extensive software development background, even if not explicitly labeled "Business Analyst," provides the foundational skills and experience to perform the BA duties, especially when the job description emphasizes technical understanding and collaboration with IT. The model should look for evidence of understanding business requirements, translating them into technical specifications, and working with development teams, which are common in software development roles.
- **Example 29:** The model incorrectly classified this as "Good Fit" when it was "No Fit." This indicates that the model might have been too lenient in this case. The job description for a Data Analyst at Regal Rexnord is specific to manufacturing operations, SIOP, sales, customer service, sourcing, and finance, and requires experience with Power BI, Google Analytics, Adobe Analytics, Python/R, Microsoft Power Apps, and SQL/Oracle. While the candidate has data analyst experience and some of these tools, the feedback suggests a lack of alignment with the *specific domain* and potentially insufficient depth in the required tools for a "Good Fit" classification. The model needs to balance flexibility with strictness, especially when domain-specific experience is implied or explicitly stated.

By incorporating these points, the model should be able to make more accurate and nuanced classifications.

Prompt Engineering

Matei Zaharia
@matei_zaharia

Prompt optimization is becoming a powerful technique for improving AI that can even beat SFT! Here are some of our research results with GEPA at Databricks, in difficult Agent Bricks info extraction tasks. We can match the best models at 90x lower cost, or improve them by ~6%.



5:59 AM · Sep 25, 2025 · **104.2K** Views

Strukturierte Extraktion

- Schätzungen zufolge liegen 80% von Unternehmensdaten unstrukturiert vor
- Übersetzung in strukturierte Formate ist ein langwieriger und aufwendiger Prozess
- Bereits sehr kleine SLMs eignen sich aber perfekt für diese Aufgabe

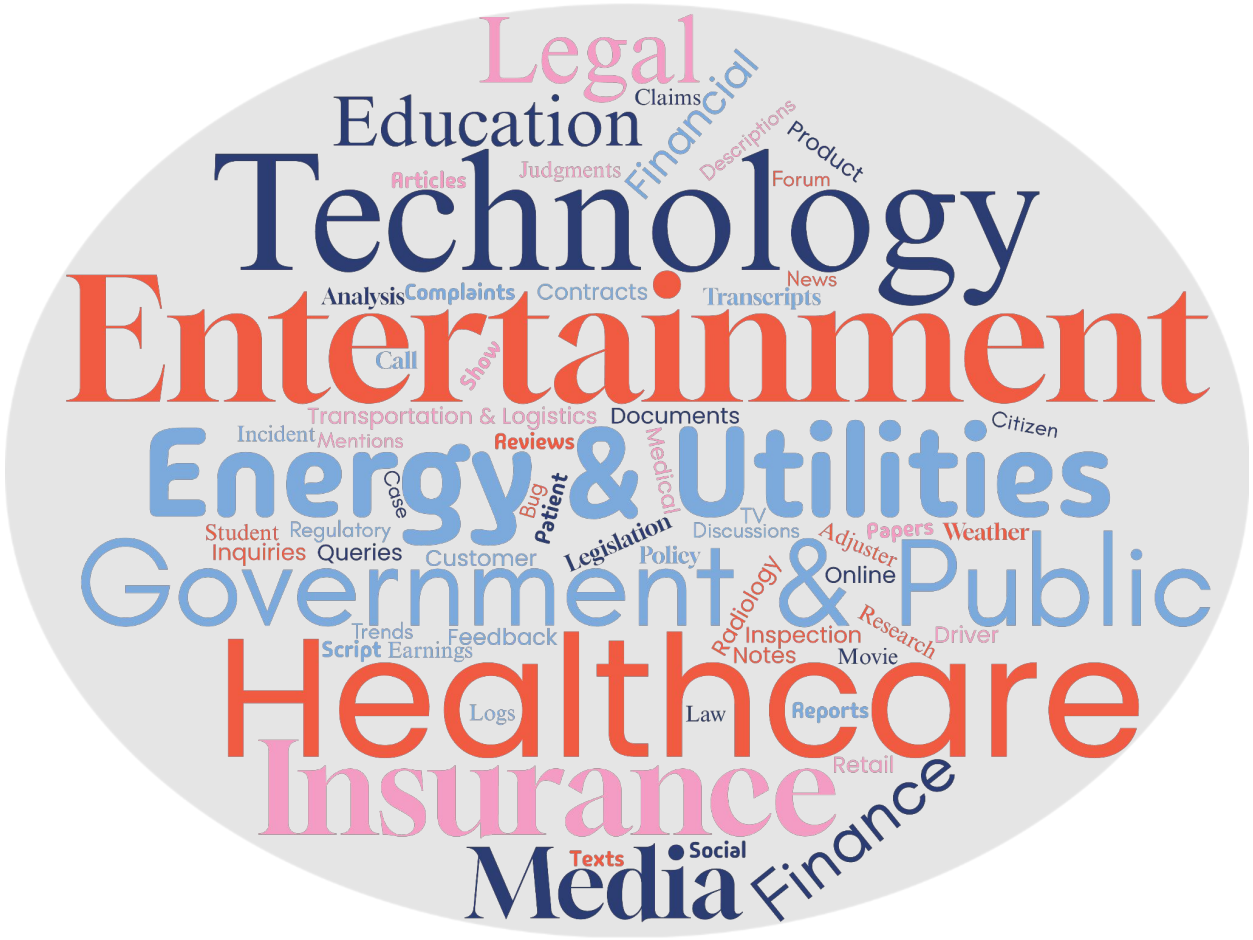
- Oft liegen unstrukturierte Daten aber in verschiedenen Formaten oder gar nur graphisch vor
- **Die Lösung:**


```
        Dear Parasol Insurance,

        My name is Aaron Bowman, and I am writing to file a claim for a recent car accident that occurred on 2024-
01-02,
        at approximately 11:45 AM. My policy number is AC-584790380.

        The accident took place at the intersection of Maple Rd and Main St. I was driving my vehicle, a maroon Ni
ssan Elantra with license plate
        number 285 3YT. At the same time, another vehicle, a silver Honda Traverse with license plate number 1DZ S
60,
        collided with my car. The driver, Douglas Small, failed to adhere to traffic rules, resulting in damage to
both vehicles.

        I promptly exchanged information with the other driver and took photos of the accident scene, including da
mages to both vehicles.
        Attached to this email are the photos, a copy of the police report, and the estimate for the repair costs.

        Kindly assist in processing this claim and let me know the next steps. You can reach me at (909)412-1229 o
r bestrada@chang.com.

        Thank you for your assistance.

        Sincerely,
        Aaron Bowman
        Unit 4232 Box 8906, DPO AE 18730
```

```
Customer(
    name='Aaron Bowman',
    policy_number='AC-584790380',
    telephone_number='(909)412-1229',
    email_address='bestrada@chang.com',
    address='Unit 4232 Box 8906, DPO AE 18730'
)
```

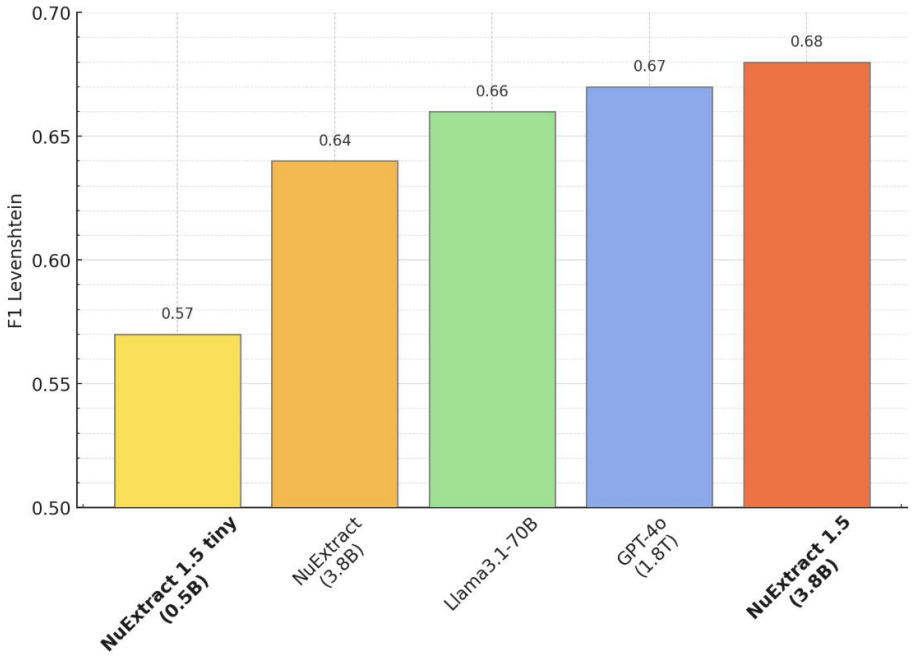
Anwendungsfall 1: Structured Extraction

Anwendungsfälle



Structured Extraction with NuExtract 1.5 tiny

NuExtract-tiny-v1.5 is a fine-tuning of Qwen/Qwen2.5-0.5B, trained on a private high-quality dataset for structured information extraction.



Zero-shot results on the structured extraction benchmark. Average F1-score over extraction problems. NuExtract 1.5 is better than NuExtract and slightly better than GPT-4o.

Lot de 10 assiettes Happy - fuchsia, en carton, mesurant 22,5 cm de diamètre.

Assiettes anniversaire - vert anis

Pour l'occasion, nous vous proposons, ici, un lot de 10 assiettes Happy, couleur fuchsia, en carton. Ces assiettes illustrées du mot Happy en doré, mesurent 22,5 cm de diamètre

Input text

```
{
  Product Information: {
    Product Name: ,
    Color: ,
    Material: ,
    Size: ,
    Quantity:
  },
  Description: ,
  Related Products: [
    {
      Product Name: ,
      Quantity:
    }
  ]
}
```

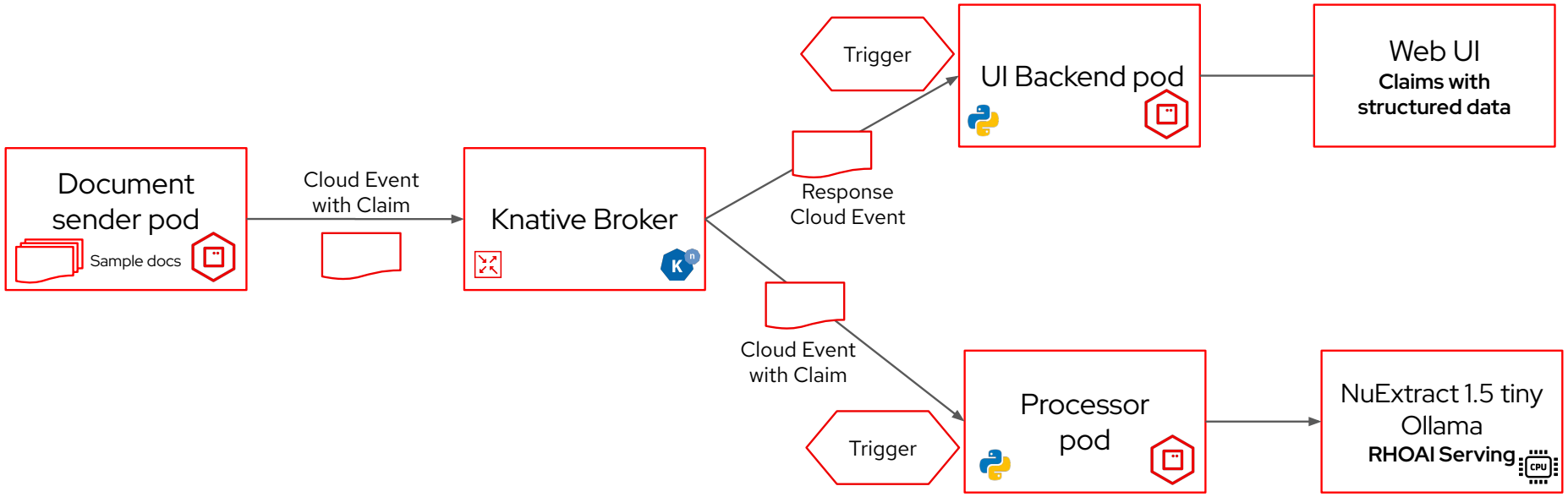
template

```
{
  Product Information: {
    Product Name: assiettes Happy ,
    Color: fuchsia,
    Material: carton,
    Size: 22,5 cm,
    Quantity: 10
  },
  Related Products: [
    {
      Product Name: assiettes anniversaire - vert ani,
      Quantity:
    }
  ]
}
```

output

Training example with a French document and an English template.

Runtime Architecture



Key Takeaways

1. Unstructured text streams are common across various industries and often contain valuable information that can be leveraged for decision-making, analytics, or operational efficiency.
2. Extracting structured information from these streams can be highly beneficial.
3. The demo showcases structured information extraction with small language models (SLM).
4. SLM use less resources, can be deployed in edge environments, use less energy, and do not require powerful GPUs.

~~Attention~~ CPU is all you need

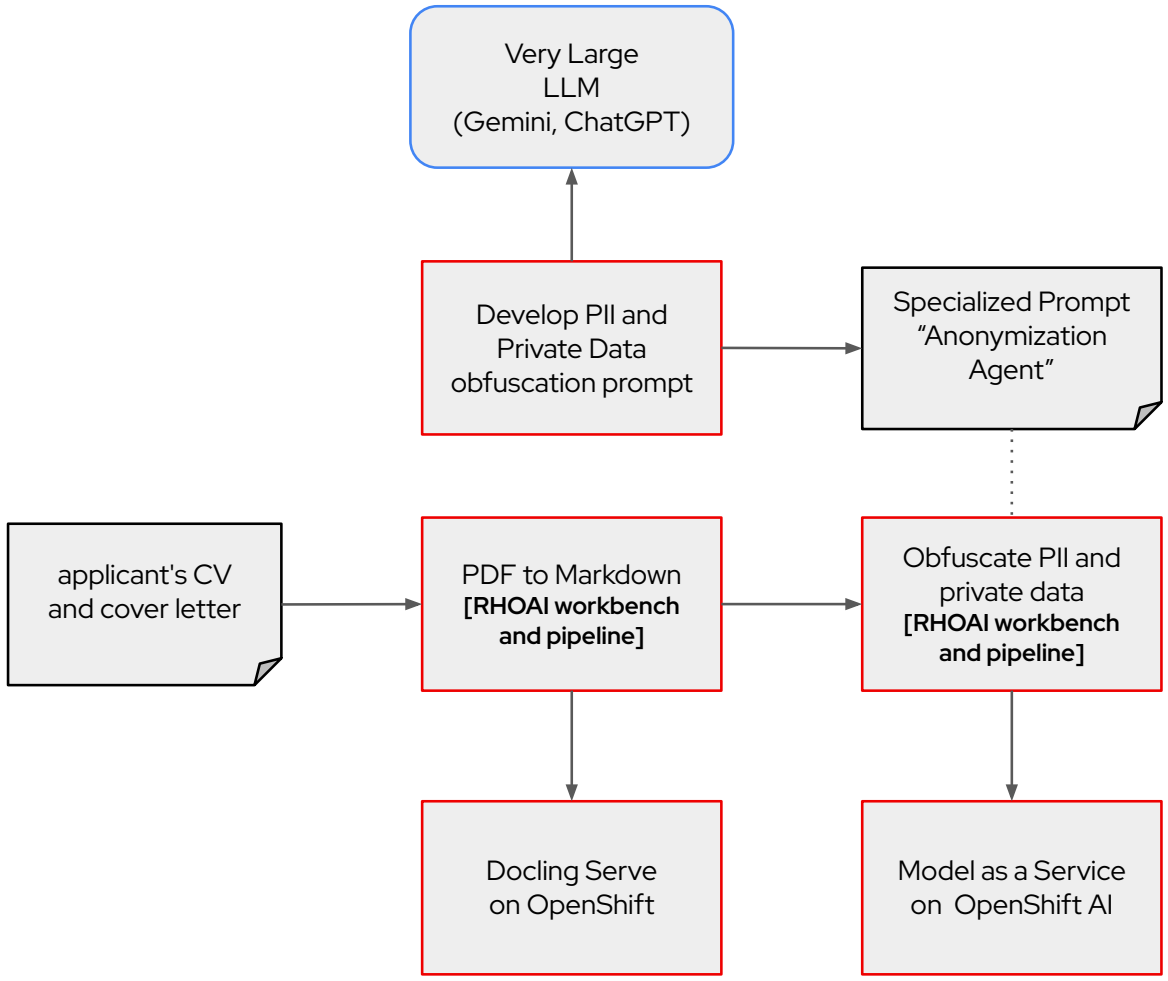
Anwendungsfall 2: Prompt Engineering

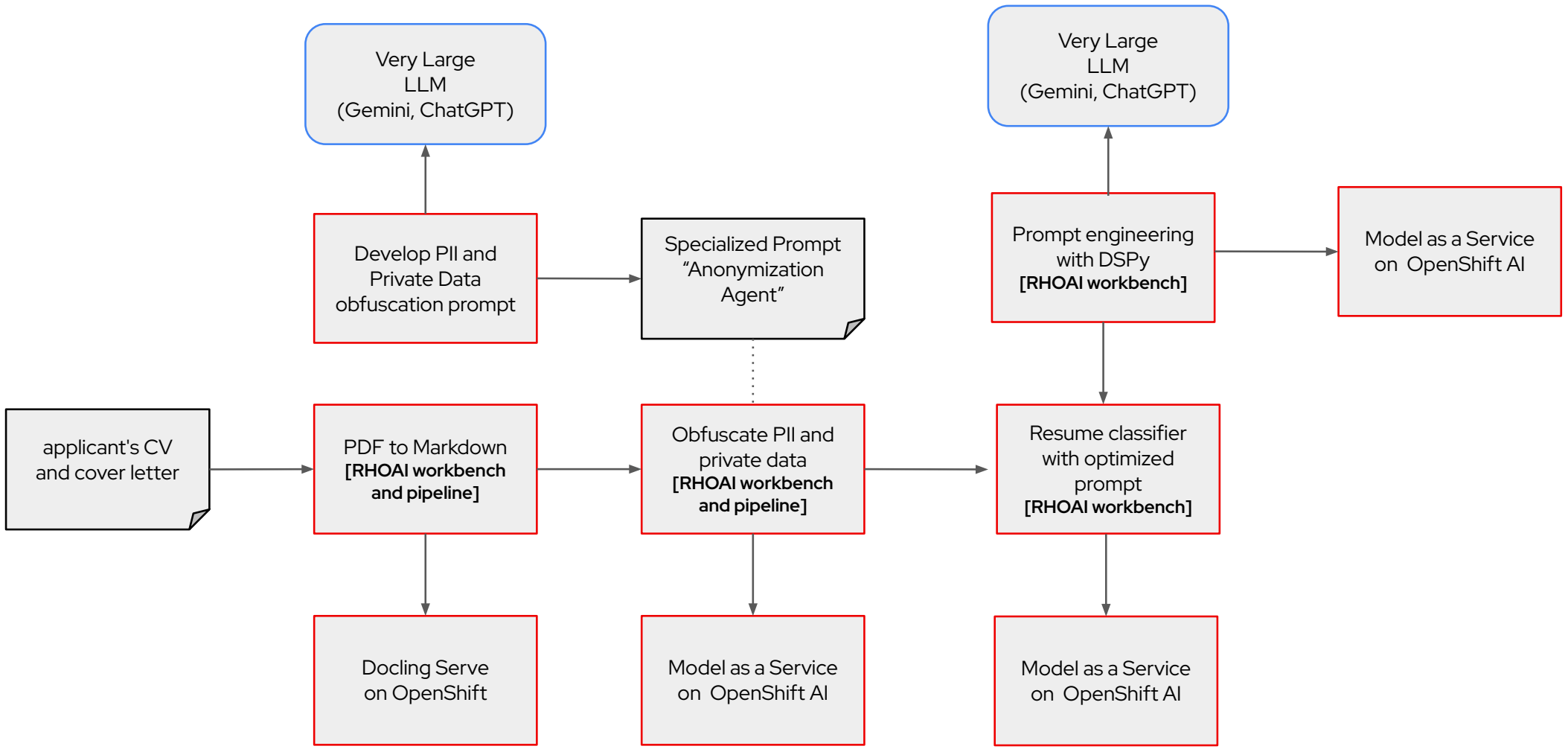
The Problem:

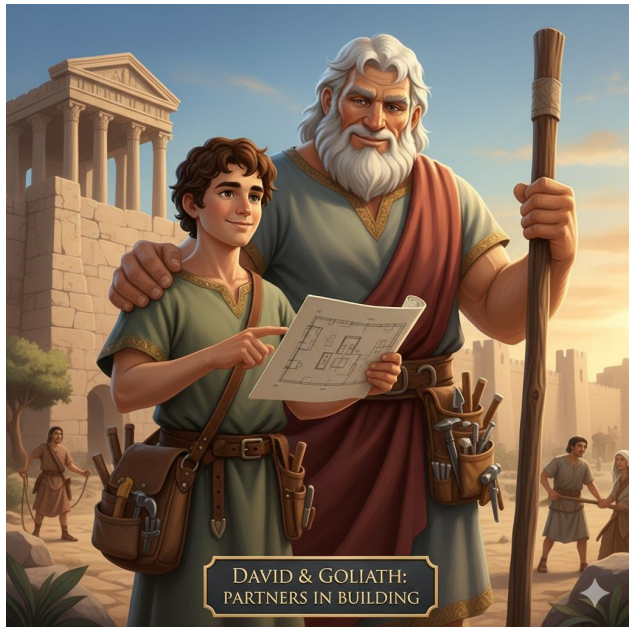
- Automated document processing in application and hiring processes is common practice.
- How can you assure a fair hiring process in your company without any bias?

The Solution:

- Obfuscate Personally Identifiable Information and Highly Sensitive Private Data from applicant's documents (CV, cover letter, etc)
- Use a fair and sophisticated matching process for pre-selecting applicants
- Data driven prompt generation
- Showcase how small language models can be used in cooperation with larger LLMs


Very Large
LLM
(Gemini, ChatGPT)
Develop PII and
Private Data
obfuscation prompt
Specialized Prompt
“Anonymization
Agent”
applicant's CV
and cover letter
PDF to Markdown
[RHOAI workbench
and pipeline]
Obfuscate PII and
private data
[RHOAI workbench
and pipeline]
Docling Serve
on OpenShift
Model as a Service
on OpenShift AI


Very Large
LLM
(Gemini, ChatGPT)
Develop PII and
Private Data
obfuscation prompt
Specialized Prompt
“Anonymization
Agent”
Very Large
LLM
(Gemini, ChatGPT)
Prompt engineering
with DSPy
[RHOAI workbench]
Model as a Service
on OpenShift AI
applicant's CV
and cover letter
PDF to Markdown
[RHOAI workbench
and pipeline]
Obfuscate PII and
private data
[RHOAI workbench
and pipeline]
Resume classifier
with optimized
prompt
[RHOAI workbench]
Docling Serve
on OpenShift
Model as a Service
on OpenShift AI
Model as a Service
on OpenShift AI



- Small models perform very well with good prompts
- Manual and automatic prompt generation with very large LLM (Gemini, ChatGPT)
- Data driven prompt generation is very powerful and avoids the manual and complex effort for prompt engineering, required to be done for each LLM individually
- All powered by OpenShift and OpenShift AI

Anwendungsfall 3: Legacy Contract Migration

What

Key metadata needed to be extracted from over 32,000 supplier contracts in various languages and formats in order to migrate documents into a new Contract Lifecycle Management System.

Why

Reviewing all the contracts and extracting the required metadata from each document would have taken months to complete, occupying associates' time that could have been spent on more strategic work.

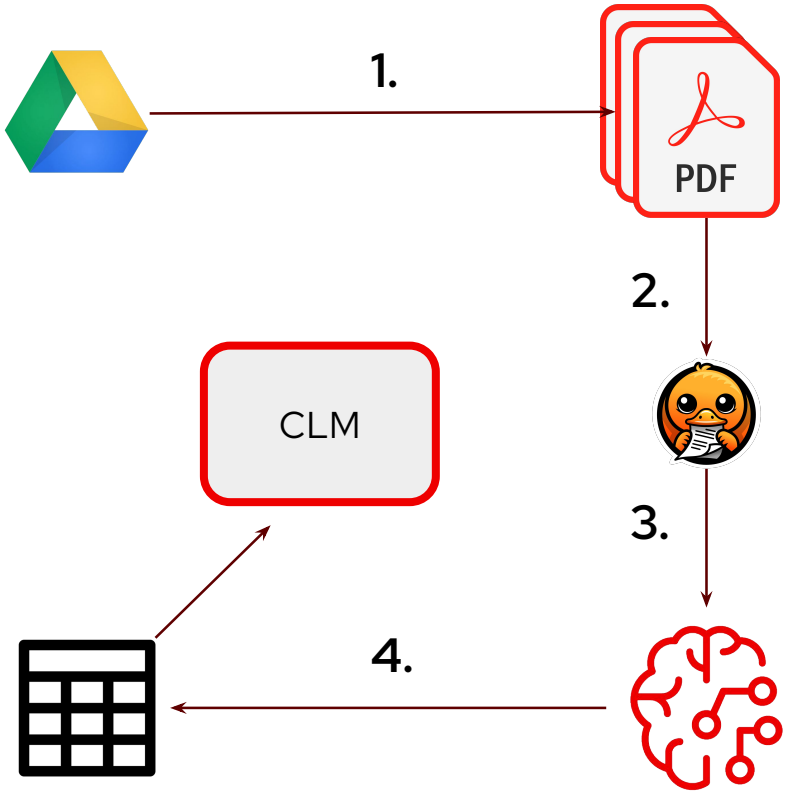
How

Leveraging AI and automation on OpenShift AI to read each contract and extract the required metadata into a structured template that is uploaded into the new CLM.

Migration Process Overview

The automation in OpenShift:

1. Fetches contract PDFs from Google Drive
2. Converts each pdf to text using Docling
3. Passes each contract's text through AI Language Models to extract metadata
4. Formats and stores the contract, metadata, and review tracking in a Google Sheet, ready to be shared with the stakeholders and CLM for review



Docling and AI models are all hosted on Red Hat OpenShift AI

Manual Extraction

14,000 hours

manual extraction and
manual review

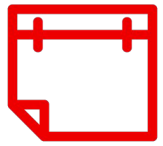
$700,000

cost of manual extraction
and manual review

>1 year

4+ FTE

1-time process
and change



AI-driven Extraction

1,700 hours

development, automation,
and manual review

$80,000

cost of development,
automation, and manual review

3 months

1 data scientist and 1 data engineer

Reusable
automation tool

Red Hat Summit

Connect

Thank you


linkedin.com/company/red-hat


facebook.com/redhatinc


youtube.com/user/RedHatVideos


twitter.com/RedHat

Jetzt Session bewerten!

Einfach QR-Code scannen, Session aus der Liste wählen und bewerten. **Vielen Dank!**

red.ht/rhsc-darmstadt-feedback

Appendix

Introducing Docling

- 🗂️ Parsing of multiple document formats incl. PDF, DOCX, XLSX, HTML, images, and more
- 📑 Advanced PDF understanding incl. page layout, reading order, table structure, code, formulas, image classification, ...
- 🧬 Unified, expressive DoclingDocument representation format
- ↪️ Various export formats (Markdown, HTML, JSON)
- 🔒 Local execution for sensitive data and air-gapped environments
- 🤖 Many plug-and-play ecosystem integrations
- 🔍 Extensive OCR support for scanned PDFs and images
- 🥚 Support of Visual Language Models
- 💻 Simple and convenient CLI