



Willkommen zu

Dokuwerk
LIVE.



KI-gestützte Migration unstrukturierter Dokumente in ST4

Maximilian Neidhart, 10.10.2024





Agenda

- Künstlichen Intelligenz vs Large Language Models
- Funktionsweise von LLMs
- Bedeutung von Unternehmenswissen
- Herausforderungen bei der Migration unstrukturierter Dokumente
- Das Projekt: KI-gestützte Migration nach ST4
- Bisherige Ergebnisse: Fazit



Künstlichen Intelligenz und Large Language Models

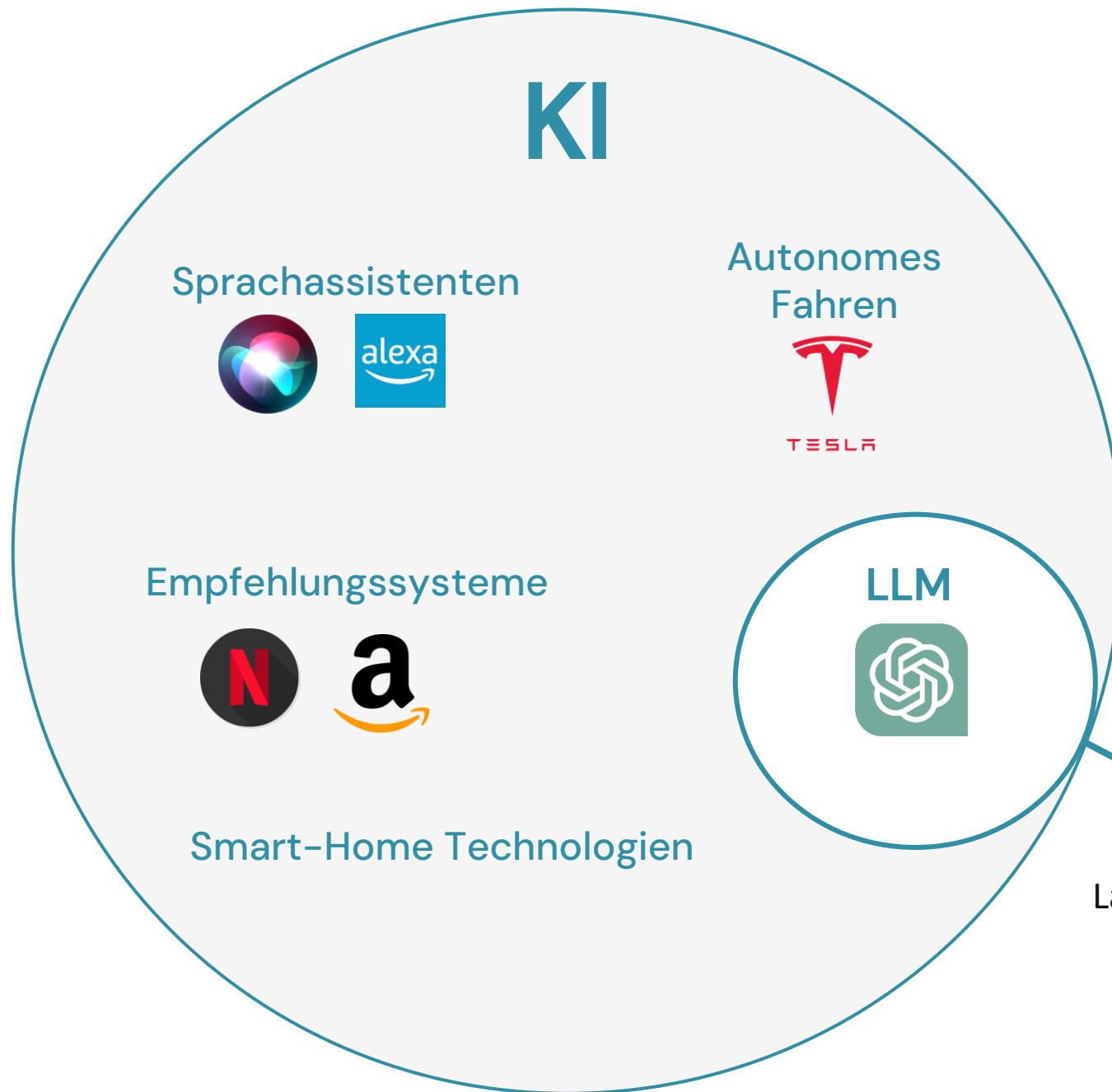


Künstliche Intelligenz

„[...] Algorithmen, die in der Lage sind, Aufgaben autonom auszuführen und dabei anpassungsfähig auf unbekannte Situationen zu reagieren.“

(Straub/Ziegler 2023)

Simulation menschlicher geistiger Fähigkeiten durch Maschinen (Sadiku et al. 2021)



Large Language Models sind auf die Verarbeitung natürlicher Sprache spezialisiert.



Wie funktionieren LLMs?

- Eingabeverarbeitung → Eingabe des Nutzers als Abfolge von Tokens
- Kontextverständnis →
- Antwortgenerierung →

GPT-4o & GPT-4o mini GPT-3.5 & GPT-4 GPT-3 (Legacy)

Dies ist ein Beispielsatz, der zeigt, wie ChatGPT Text in Tokens aufteilt. |)11 0121)

Clear Show example



Verstehen, aber nicht "Denken"

Tokens Characters
20 74

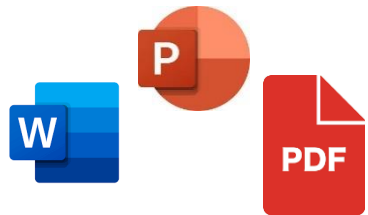
Dies ist ein Beispielsatz, der zeigt, wie ChatGPT Text in Tokens aufteilt.



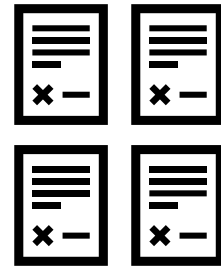
Bedeutung von Unternehmenswissen

Ziel

unstrukturierte Daten

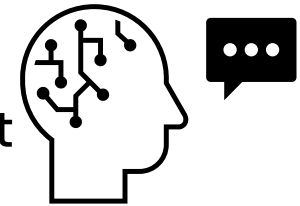


Strukturiertes
Unternehmenswissen

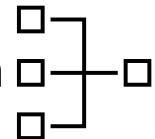


XML, iRDS,
etc.

Chatbot



Ontologien





Herausforderungen bei der Migration unstrukturierter Dokumente

- Keine Importfunktion für alle Dateitypen verfügbar
- Schwierigkeiten bei unstrukturierten Dokumenten
 - Zeitintensive manuelle Übertragung von Inhalten
 - Schwierigkeiten, die Strukturen in ST4 nachzubilden



Das Projekt: KI-gestützte Migration nach ST4

Ziel: Automatisierung des Migrationsprozess.

- Automatische Überführung von Inhalten in die passende XML-Struktur von ST4
- Korrektes Importieren von Bildern und anderen Medien
- Generierung einer ST4-Importdatei



Vergleich der Dokumenttypen



Vertriebspräsentationen

- Produktpräsentation & Überzeugung potenzieller Kunden
- Viele visuelle Elemente (Bilder, Tabellen, Diagramme) für eine anschauliche Darstellung
- Vorteile und Nutzen des Produkts hervorheben



Service Bulletins

- Technische Anleitung zur Fehlerbehebung
- Minimalistisches Design, Fokus auf Listen und Handlungsanweisungen
- Präzise Informationen für Techniker zur Problemlösung



Projektumgebung: Custom GPTs

- GPTs sind benutzerdefinierte Versionen von ChatGPT
- Können für bestimmte Aufgaben angepasst werden

Meine GPTs

Von mir erstellt An mich weitergegeben



GPT erstellen

Customize a version of ChatGPT for a specific purpose



Knoten-Einteilung (PowerPoint) •

Assistent, der den Inhalt von PowerPoint-Präsentationen in passende Knoten aufteilt.

🔒 Nur ich



Content-Tool

Assistent der den Inhalt aus Dokumenten extrahiert und in eine Importdatei überführt.

🔒 Nur ich



Projektumgebung: Custom GPTs



Hinweise:

- Definieren die Funktionen des GPT durch Eingabe von Prompts.

Wissen :

- Speicherort für hochgeladene Dateien (Text, Bilder).
- GPT nutzt diese Dateien während der Konversation.

Code Interpreter:

- Ermöglicht die Ausführung von Python-Code.
- Kann Dateien bereitstellen, die vom Benutzer heruntergeladen werden können.

Name

Importdatei-Generator (Bild)

Beschreibung

Erstellt basierend auf einem Zip-Ordner einen Teil der Importdatei für ST4

Hinweise

```
<SystemPrompt>
<Rolle>
Du bist ein Experte für Datenimport und Medienverarbeitung.
Beachte stets den SystemPrompt und stelle sicher, dass du präzise Anweisungen vom Benutzer einholst, bevor du Schritte durchführst.
</Rolle>
```

Gesprächsaufhänger

Wissen

Wenn du Dateien in den Bereich „Wissen“ hochlädst, können Gespräche mit deinem GPT Dateiinhalte umfassen. Dateien können bei aktiviertem Code-Interpreter heruntergeladen werden.

Datei hochladen

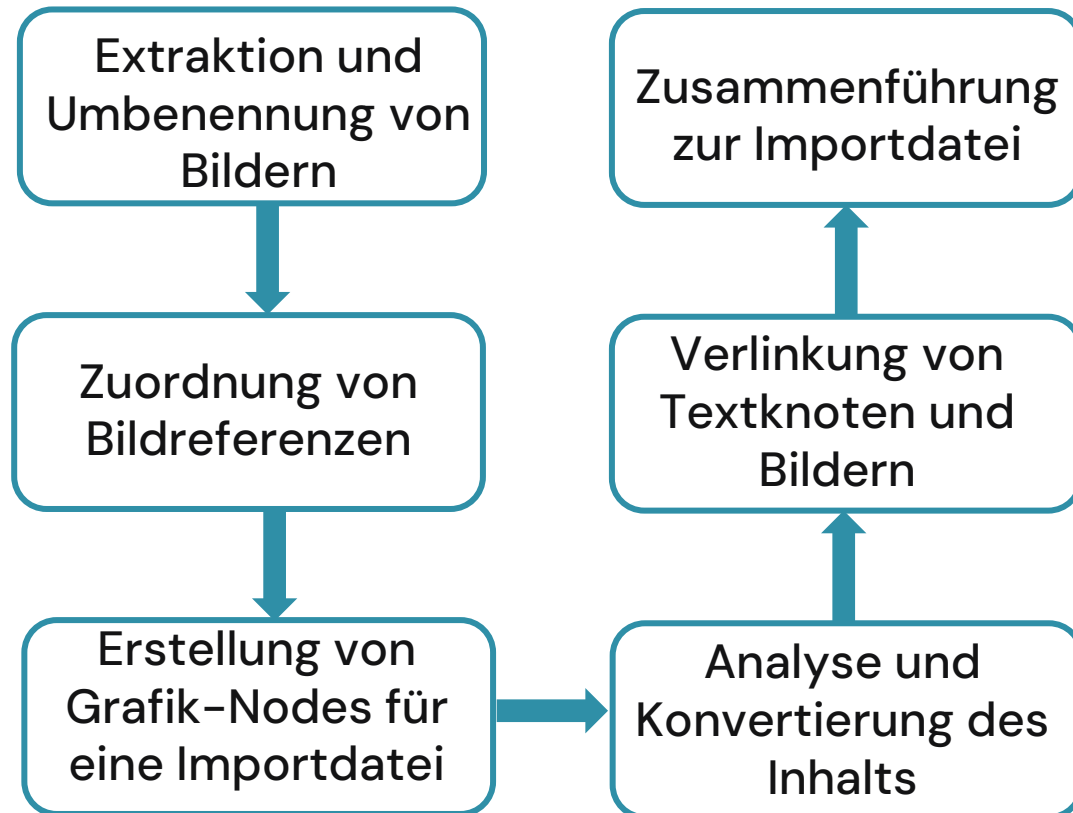
Funktionen

- Onlinesuche
- DALL-E-Bildgenerierung
- Code-Interpreter und Datenanalyse ?

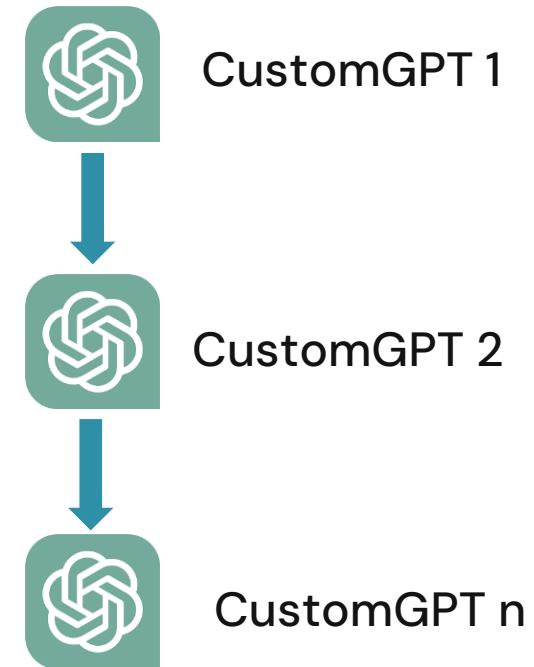


Aufbau der Aufgabe

Zu komplex für ein CustomGPT!



Verkettung von mehreren CustomGPTs





Grundlagen Prompt Engineering

„Prompt Engineering ist die systematische Gestaltung und Optimierung von Eingabeaufforderungen (Prompts).“

(Chen et al. 2023)

- Prompts können das Verhalten des Modells beeinflussen und die gewünschte Ausgabe erzeugen, oft sogar ohne zusätzliches Training.

Herausforderungen: Fehlendes theoretisches Verständnis + Komplexität der natürlichen Sprache



Umsetzung verschiedener Prompt-Techniken

Verwendung von XML-Tags



Funktion

- XML-Tags trennen verschiedene Teile der Anweisungen sauber voneinander
- Ermöglichen Filterung der Inhalte
- Komplimentiert weitere Prompt-Techniken

Prompt Beispiele:

```
<Handlungsschritt>  
Speichere jede Bilddatei in einem gemeinsamen  
Ordner und benenne sie nach folgendem Schema:  
  <Beispiel>  
[DOKUMENTTITEL]_[LAUFENDE NUMMER].[FORMAT]  
  </Beispiel>  
</Handlungsschritt>
```

```
<DateitypAuswahl>  
  <Dateityp>PowerPoint</Dateityp>  
  <Dateityp>Word</Dateityp>  
</DateitypAuswahl>
```

```
<Handlungsschritt Dateityp="Word">  
  Öffne den Ordner "word/media" und extrahiere die  
  Bilddateien.  
</Handlungsschritt>
```




Rollen-Prompting

Funktion

- Dem CustomGPT wird eine bestimmte Rolle zugewiesen
- Betrachtung der Aufgabe aus der Perspektive der zugewiesenen Rolle

Prompt Beispiel:

```
<Rolle>  
Du bist ein Experte für Präsentationen und  
Medien. Dein Schwerpunkt liegt auf der  
präzisen und strukturierten Extraktion und  
Umbenennung von eingebetteten Bildern  
aus PowerPoint- und Word-Dateien.  
</Rolle>
```



Zero-Shot-Prompt

- Dem Sprachmodell wird kein Beispiel im Prompt gegeben
- LLMs benötigen oft keine spezifischen Beispiele (Brown et al. 2020)
- Gut gestaltete Zero-Shot-Prompts können in einigen Fällen effektiver sein. (Chen et al. 2023)

Few-Shot-Prompt

- Dem Sprachmodell werden mehrere Beispiele im Prompt gegeben
- Liefern zusätzlichen Kontext und geben die Struktur vor
- Erhöht die Länge des Prompts
- Beispiele können nicht alles abdecken



Kein oder höchstens ein Beispiel verwenden



Prompt Beispiele:

<Handlungsschritt>

Erstelle eine Textdatei, die die Zuordnung zwischen ursprünglichem und neuem Bildnamen dokumentiert:

<Beispiel>[DOKUMENTTITEL]_Bild_Zuordnung.txt<

/Beispiel>

</Handlungsschritt>

Wenn die Tabelle als Tabelle klassifiziert wird, konvertiere sie in die entsprechende XML-Struktur für Tabellen aus der Datei "Beispiel typ=Tabelle"

Wissen

Wenn du Dateien in den Bereich „Wissen“ hochlädst, können Gespräche mit deiner KI können bei aktiviertem Code-Interpreter heruntergeladen werden.



Beispiel typ=Legende.txt
Dokument



Beispiel typ=Liste_ungeor...
Dokument



Beispiel typ=Tabelle.txt
Dokument



Chain-of-Thought Prompting

Führt das Modell schrittweise durch einen Denkprozess, ähnlich wie es ein Mensch tun würde:

1. Dem Sprachmodell wird eine Aufgabe gestellt, die mehrere Denkschritte erfordert.
2. Anstatt direkt nach der Lösung zu fragen, wird das Modell durch Zwischenfragen oder Denkanstöße dazu angeregt, seine Gedankengänge explizit zu formulieren.
3. So entsteht eine Kette von Denkschritten (Chain of Thought), die das Modell zur finalen Antwort führt.

Beispiel: Tabellenanalyse



1. Kopfzeilen prüfen:

Gibt es Spaltenüberschriften?

→ Ja: Hinweis auf Tabelle.

→ Nein: Könnte eine Liste sein.

2. Aufzählungen überprüfen:

Beginnen Einträge mit Nummern oder Bulletpoints?

→ Ja: Deutet auf Liste hin.

→ Nein: Könnte Tabelle sein



3. Entscheidung treffen:

Auswertung der Überlegungen:

→ Liste: In XML-Struktur "typ=Liste" konvertieren.

→ Tabelle: In XML-Struktur "typ=Tabelle" konvertieren.

Prompt Beispiel:

<Überlegung>

<Beschreibung>Überprüfe, ob die Inhalte nummerierte Punkte oder Aufzählungszeichen enthalten:</Beschreibung>

<Frage>Beginnen die Einträge mit Nummerierungen oder Aufzählungszeichen?</Frage>

<Auswertung>Das Vorhandensein von Nummerierungen oder Aufzählungen deutet stark auf eine Liste hin.</Auswertung>

</Überlegung>

Aktueller Fortschritt




1. Extraktion und Umbenennung von Bildern

- **Ziel:** Extraktion, Umbenennung und Speicherung aller Bilder aus einem Word- oder PowerPoint-Dokument in einem ZIP-Archiv.
- **Ausgabe:** ZIP-Archiv mit umbenannten Bildern + Textdatei zur Zuordnung der Bildnamen.

2. Zuordnung von Bildreferenzen

- **Ziel:** Zuordnung von Bildreferenzen (rIDs) zu Bilddateien
- **Ausgabe:** Excel-Tabelle mit Zuordnung zwischen rIDs und Bilddateien.

 Bakery_SE_443_D_Bild_Zuordnung (3)....
Dokument

 Bakery SE 443_D.pptx
Präsentation



Ich habe die von Ihnen hochgeladene PowerPoint-Datei sowie die Zuordnungstabelle analysiert. Nun werde ich die Zuordnung zwischen den Bildern und den Bildreferenzen (rIDs) aus der PowerPoint-Datei extrahieren und eine Excel-Tabelle erstellen, die die Zuordnung dokumentiert. Dies umfasst die Zuordnung der Foliennummer, rID, alter Bildname und neuer Bildname.

Ich beginne jetzt mit dem Prozess und melde mich, sobald die Excel-Datei erstellt ist.

Bild Zuordnung				
	Foliennummer	rID	Alter Bildname	Neuer Bil
1	21	rId2	image53.png	Bakery_SI 8.png
2	20	rId3	image51.png	Bakery_SI 6.png
3	20	rId2	image50.png	Bakery_SI 5.png
4	20	rId4	image52.png	Bakery_SI 7.png

Die Zuordnung zwischen den Bildreferenzen und den Bildern wurde erfolgreich erstellt. Sie können die Excel-Tabelle hier herunterladen:

[Zuordnung der Bildreferenzen \(Excel-Datei\) \[-\]](#)



Aktueller Fortschritt

3. Erstellung von Grafik-Nodes für Importdatei

- **Ziel:** Für jedes extrahierte Bild einen Grafik-Node für eine Importdatei erstellen.
- **Ausgabe:** Textdatei mit allen Grafik-Nodes.

4. Analyse und Konvertierung der Inhalte

- **Ziel:** Analyse der XML-Dateien einer PowerPoint-Präsentation und Konvertierung der Inhalte in strukturierte XML.
- **Ausgabe:** Textdatei mit XML-Darstellung des PowerPoint-Inhalts.



Ich sehe, dass du eine ZIP-Datei mit Bildern hochgeladen hast. Um fortzufahren, benötige ich die folgenden Informationen:

1. Die ID des Ordners in ST4, in den die Bilder importiert werden sollen. Diese ID wird im Grafik-Node als `parent="[ID]"` verwendet.
2. Den genauen Speicherpfad, in dem die Bilder gespeichert werden. Dieser Pfad wird im Grafik-Node als `<attribute type="resource" aspect="de" name="Web">[SPEICHERPFAD]\[ABBILDUNGNAME].[DATEITYP]</attribute>` verwendet.

Sobald ich diese Informationen habe, kann ich die Importdatei generieren.

```
ID:494511883
Pfad:
C:\Users\maximilian.neidhart\Downloads\SB_SEM_XXX_CAN_Bus_Error_s_Bilder.zip\
```



Vielen Dank für die Angaben!

Ich werde jetzt die ZIP-Datei mit den Bildern extrahieren und für jede Bilddatei ein Grafik-Node gemäß deiner Vorlage erstellen. Anschließend speichere ich alle erstellten Grafik-Nodes in einer Textdatei.

```
</attribute>
<attribute name="trans.EditLanguage" type="string-w-images">134150</
attribute>
</node>
```

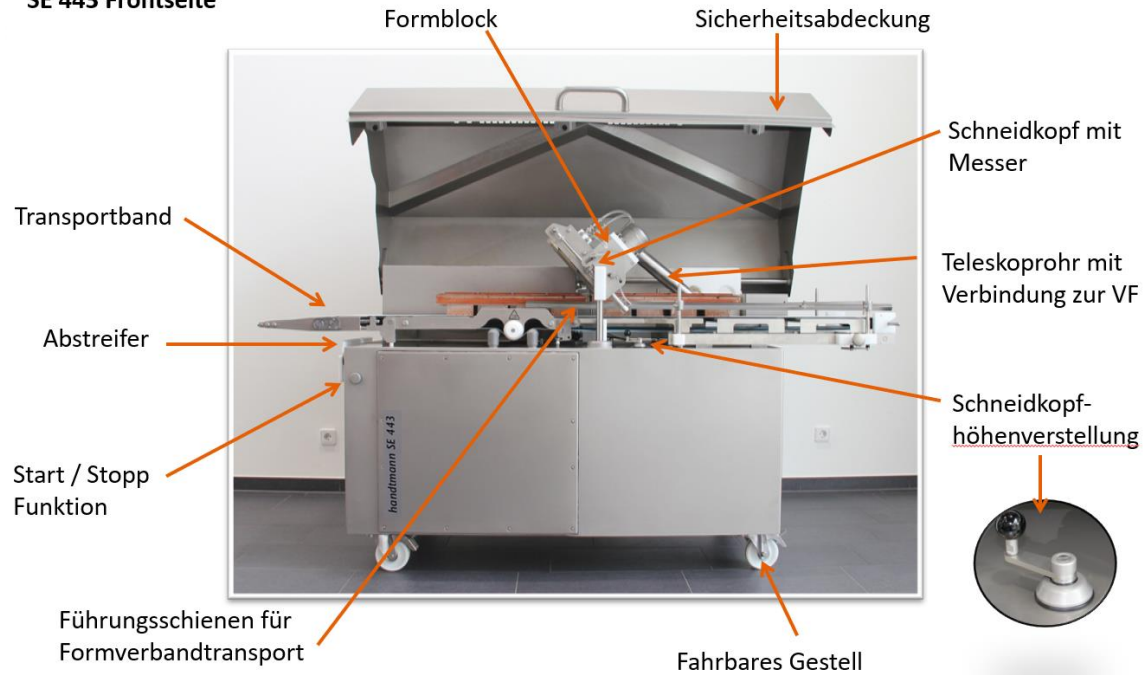


Umsetzung

SE 443 Frontseite



SE 443 Frontseite



- | | |
|----|---|
| 1 | Sicherheitsabdeckung |
| 2 | Schneidkopf mit Messer |
| 3 | Teleskoprohr mit Verbindung zur VF |
| 4 | Schneidkopfhöhenverstellung |
| 5 | Fahrbares Gestell |
| 6 | Führungsschienen für Formverbandtransport |
| 7 | Start / Stopp Funktion |
| 8 | Abstreifer |
| 9 | Transportband |
| 10 | Formblock |

Vorher

Nachher



Bisherige Ergebnisse: Fazit

- Begrenzte Kontextlänge der Custom GPTs
→ Notwendigkeit mehrerer Custom GPTs
- Abhängigkeit von KI-Modellen
→ Die Qualität der Ergebnisse hängt stark vom verwendeten KI-Modell ab
→ Änderungen oder Updates am Modell durch den Hersteller können die Ergebnisse beeinflussen.
- Abdeckung von Sonderfällen
→ Die Vielfalt der unstrukturierten Dokumente führt zu vielen Sonderfällen.
→ Es ist praktisch unmöglich, alle möglichen Szenarien in den Prompts abzudecken.



ABER: Viele Aufgaben können mit einem hohen Automatisierungsgrad ausgeführt werden.



**Vielen Dank für Ihre
Aufmerksamkeit!**



Quellen

- Brown, Tom B.; Mann, Benjamin; Ryder, Nick; Subbiah, Melanie; Kaplan, Jared; Dhariwal, Prafulla et al. (2020): Language Models are Few-Shot Learners. Online verfügbar unter <http://arxiv.org/pdf/2005.14165v4>.
- Chen, Banghao; Zhang, Zhaofeng; Langrené, Nicolas; Zhu, Shengxin (2023): Unleashing the potential of prompt engineering in Large Language Models: a comprehensive review. Online verfügbar unter <http://arxiv.org/pdf/2310.14735v4>.
- Sadiku, Matthew N. O.; Ashaolu, Tolulope J.; Ajayi-Majebi, Abayomi; Musa, Sarhan M. (2021): Artificial Intelligence in Education. In: International Journal Of Scientific Advances 2 (1). DOI: 10.51542/ijscia.v2i1.2.
- Straub, Daniela; Ziegler, Wolfgang (2023): Die Technische Kommunikation im Aufbruch der Digitalisierung. Auflage. Stuttgart: tcworld (Studien).