

Accessing Customer-Specific SAP Documentation Using Large Language Models (LLMs)

Bachelorarbeit

eingereicht von: **Martin Drzola**

Matrikelnummer: 52107951

im Fachhochschul-Bachelorstudiengang Wirtschaftsinformatik (0470)
der Ferdinand Porsche FernFH

zur Erlangung des akademischen Grades <einer/eines>

Bachelor of Arts in Business

Betreuung und Beurteilung: FH-Prof. DI Dr. Igor Miladinovic

Wiener Neustadt, Mai 2024

Ehrenwörtliche Erklärung

Ich versichere hiermit,

1. dass ich die vorliegende Bachelorarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Alle Inhalte, die direkt oder indirekt aus fremden Quellen entnommen sind, sind durch entsprechende Quellenangaben gekennzeichnet.
2. dass ich diese Bachelorarbeit bisher weder im Inland noch im Ausland in irgendeiner Form als Prüfungsarbeit zur Beurteilung vorgelegt oder veröffentlicht habe.

Wien, 30.05.2024

Unterschrift

Creative Commons Lizenz

Das Urheberrecht der vorliegenden Arbeit liegt bei beim Autor. Sofern nicht anders angegeben, sind die Inhalte unter einer Creative Commons „Namensnennung - Nicht-kommerziell - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz“ (CC BY-NC-SA 4.0) lizenziert.

Die Rechte an zitierten Abbildungen liegen bei den in der jeweiligen Quellenangabe genannten Urheber*innen.

Die Kapitel 1 bis 4 der vorliegenden Bachelorarbeit wurden im Rahmen der Lehrveranstaltung „Bachelor Seminar 1“ eingereicht und am 12.2.2024 als Bachelorarbeit 1 angenommen.

Kurzzusammenfassung: Zugriff auf kundenspezifische SAP-Dokumentation mithilfe von Large Language Models (LLMs)

Die vorliegende Arbeit untersucht die Frage, ob Large Language Models (LLMs), insbesondere OpenAIs GPT-4, den Zugang zu und die Pflege von kundenspezifischen Informationen im Kontext von oft hochkomplexen und maßgeschneiderten SAP-Implementierungen, die häufig unter Lücken und Inkonsistenzen in der Dokumentation leiden, fördern können.

Ein Prototyp-Chatbot, der mit GPT-4 betrieben wird und auf SAPUI5 basiert, wurde entwickelt, um die Interaktion von Consultants und (Key) User mit wesentlichen SAP-bezogenen Informationen zu erleichtern. Die Studie verwendete qualitative Inhaltsanalysen durch geführte Interviews mit wichtigen Stakeholdern, die Erfahrung mit SAP-Implementierungen haben, um die praktische Anwendung des Chatbots in unternehmensspezifischen Kontexten zu bewerten.

Die Ergebnisse zeigen, dass LLMs erheblich dazu beitragen können, das Verständnis zu SAP-Implementierungen zu verbessern, jedoch ohne die statische Dokumentation vollständig ersetzen zu können. Die Studie betont auch die Bedeutung der kontinuierlichen Verbesserung, um die Fähigkeiten von LLMs vollständig auszuschöpfen, insbesondere zur Pflege bestehender Dokumentation.

Diese Arbeit liefert wertvolle Einblicke in die Rolle generativer KI-Technologien im ERP-Systemmanagement und zeigt auf, dass LLMs wie GPT-4 kritische Dokumentationslücken überbrücken und die User Experience in komplexen SAP-Umgebungen verbessern können.

Schlagwörter:

SAP, Dokumentation, LLM, ChatGPT, Kundenentwicklungen, GPT-4

Abstract: Accessing Customer-Specific SAP Documentation Using Large Language Models (LLMs)

This thesis investigates the question of whether Large Language Models (LLMs), specifically OpenAI's GPT-4, are able to foster access to and maintenance of customer-specific information in the context of often highly complex and customized SAP implementations, which often suffer from gaps and inconsistencies in documentation.

A prototype chatbot powered by GPT-4 and based on SAPUI5 was developed to facilitate interaction of both consultants and (key) users with essential SAP-related information. The study employed qualitative content analysis through guided interviews with key stakeholders experienced in SAP implementations to evaluate the chatbot's practical application within company-specific contexts.

The findings indicate that LLMs can significantly help improving the understanding of SAP implementations, however without being able to completely replace static documentation. The study also highlights the importance of continuous model improvement to fully leverage the capabilities of LLMs, especially for maintaining existing documentation.

This thesis contributes valuable insights into the role of generative AI technologies in ERP system management, suggesting that LLMs like GPT-4 can bridge critical documentation gaps and enhance the overall user experience in complex SAP environments.

Keywords:

SAP, documentation, LLM, ChatGPT, customer developments, GPT-4

Table of Contents

1. INTRODUCTION	1
2. SAP – APPLICATION & CONSULTING	4
2.1 Application	4
2.1.1 Overview of SAP Software	4
2.1.2 Core Modules of SAP	4
2.1.3 SAP Architecture	5
2.1.4 SAP's Role in Business Processes	7
2.1.5 Advances in SAP Technologies	8
2.2 Consulting	8
2.2.1 Role of SAP Consultants	8
2.2.2 SAP Implementation Methodologies	10
3. LARGE LANGUAGE MODELS (LLMS)	13
3.1 Introduction to Large Language Models	13
3.1.1 Definition and Evolution of LLMs	13
3.1.2 Components and Architecture	14
3.2 Functionality and Capabilities of LLMs	15
3.2.1 Language Understanding and Generation	15
3.2.2 Training an LLM	16
3.3 LLMs in Business Applications	18
3.3.1 Use Cases in Various Industries	18
3.4 Challenges and Limitations of LLMs	19
3.4.1 Technical Challenges	19
3.4.2 Ethical and Privacy Concerns	20
3.5 Popular Models	22

1. Introduction	V
4. INTEGRATING LLMS WITH SAP	24
4.1 Integration Framework	24
4.1.1 Assessment of SAP System Requirements	24
4.1.2 Selection of Suitable LLMs	25
5. DESIGN OF PROTOTYPE	28
5.1 Frontend Design	28
5.2 Language Model	29
5.3 Test Design	30
5.3.1 General SAP Processes	30
5.3.2 Customer Documentation	30
6. IMPLEMENTATION OF PROTOTYPE	32
6.1 User Interface	32
6.2 Training Material	33
6.3 Operational Costs	35
7. EVALUATION	37
7.1 Justification of the Selected Method	37
7.2 Scenario for Assessment	38
7.3 Methodology: Qualitative Content Analysis	39
7.3.1 Introduction of Interview Partners	41
7.3.2 Interview Guide	42
7.3.3 Process of Transcription	43
7.3.4 Evaluation Categories	44
7.4 Results	44
7.4.1 Quality and Style of Responses	45
7.4.2 Comparison to Static Documentation	49
7.4.3 Usefulness	50

1. Introduction	VI
7.4.4Challenges and Opportunities	51
8. CONCLUSION	57
9. REFERENCES	59
10. INDEX OF ABBREVIATIONS	70
11. APPENDIX	71
11.1 Interview Guide	71
11.2 Electronic Bank Statement Customer Documentation	71
11.3 Interview Mag. Oliver Szikonya	71
11.4 Interview Mag. Marius Treske	71
11.5 Interview Christian Haberfehlner	71

1. Introduction

This thesis delves into the complex and multifaceted domain of SAP implementations, focusing on the challenges that users and consultants face, especially in the context of legacy customer implementations, many of which suffer from inadequate documentation.

The specific question this thesis aims to answer is whether OpenAI's GPT-4 can be used to facilitate access to and maintenance of customer-specific information for SAP users and consultants. This inquiry is motivated by the hypothesis that GPT-4 can indeed be successfully employed to facilitate access to and maintenance of such information, suggesting that Large Language Models (LLMs) could effectively bridge the gap in SAP documentation.

ERP systems are crucial to the seamless operation of global business infrastructures and SAP, as a leading provider of such solutions, offers a robust framework for businesses to integrate diverse functions—from finance and human resources to supply chain management and customer relationship management—into a cohesive and efficient system. However, the implementation of such a comprehensive system is not without its challenges. As has long been acknowledged, one of the primary hurdles in SAP implementation is its inherent complexity. This complexity is not just technical but also organizational, as it necessitates a deep understanding of varied business processes and their interdependencies. (Holub and Bruckner 2016)

Moreover, the nature of SAP projects, often spanning several years and evolving through various business and technological changes, leads to a proliferation of customer-specific customizations and developments. These adaptations to the standard software and processes, while necessary for addressing unique business needs, add layers of complexity, particularly when they are poorly documented or, as is often the case, undocumented.

In many cases, however, documentation does exist but is very hard to find and/or understand, rendering it useless. This lack of documentation creates a significant challenge, not only for ongoing maintenance and upgrades but also for new

consultants and users – who often have very different expectations and needs (Schmidt 2005) – attempting to navigate these systems.

Those challenges associated with SAP implementations, however, are not unique to SAP systems alone but are reflection of broader issues that pervade many IT-related projects. In the realm of software development and IT projects in general, similar complexities and documentation gaps are commonplace. These challenges stem from the dynamic nature of technology, evolving business requirements, and the need for solutions to cater to often highly specific organizational needs. (Martyn and Vickery 1970)

However, the specificities and intricacies of SAP systems, characterized by their vast scope and deep integration into most if not all of the core business processes, often necessitate unique solutions to these challenges. The highly customized nature of many SAP environments, coupled with the critical role they play in an organization's operations, means that any solution to address these challenges needs to be tailored to the SAP context.

The introduction of LLMs in the context of this research presents a novel and potentially transformative solution. This thesis explores how LLMs can enhance access to and management of customer-specific information. Given the novel application of these models, the research question is not only intriguing but also highly relevant to numerous companies and organizations. The implications of this study may significantly influence how businesses interact with and utilize SAP documentation.

By providing an intuitive and intelligent means of accessing customer-specific information, LLMs have the potential to alleviate some of the most pressing challenges faced by users and consultants alike, such as understanding legacy customizations or developments, identifying best practices for specific business processes, and troubleshooting issues more efficiently.

This exploration is situated at the intersection of technology and business process management, underscoring the evolving landscape of ERP system implementation and support. Through a meticulous examination of the use of Large Language Models in accessing and understanding customer-specific SAP documentation via a functioning prototype, this thesis aims to contribute not only valuable insights

to both the academic field and the practical realm of SAP consultancy and usage, but also a practical foundation on which future innovations and solutions can build. The ultimate objective is to delineate a path toward more streamlined, accessible, and user-friendly approaches to managing the complexities inherent in SAP systems, particularly those with significant legacy customizations and documentation gaps.

In pursuing this investigation and examining the current state of research of the relevant topics, the thesis will first establish a foundational understanding of the complexities inherent in SAP systems, followed by an in-depth analysis of the specific challenges posed by legacy SAP implementations and inadequate documentation. Subsequently, it will delve into the potential of Large Language Models as a tool for addressing these challenges, exploring both theoretical underpinnings and practical applications. The culmination of this study is the development of a proof-of-concept model and the assessment of its merit, demonstrating the practical application and efficacy of Large Language Models in enhancing the efficiency, accessibility, and overall effectiveness of SAP system implementations and maintenance.

2. SAP – Application & Consulting

2.1 Application

2.1.1 Overview of SAP Software

SAP has long dominated the ERP software industry with the largest market share (Kerim Goztepe, Muammer Karaman, and Hayrettin Catalkaya 2015), and it continues to be regarded as the leading provider of on-premise ERP installations. Additionally, SAP's standing in the cloud ERP market is noteworthy, consistently ranking among the top contenders.

Despite experiencing only slow growth in its first decade following its establishment in 1972, SAP already proved to be a global player with the very first client being ICI, a British chemical firm. SAP also benefited from several factors including the Year 2000 problem, the move to client/server computing and growing sentiment to buy software rather than build it and the fact that renowned consulting firms including Andersen Consulting (later Accenture), Ernst & Young (EY), Deloitte and PwC provided comfort and security to potential customers (Mirchandani 2014).

Introduced as a cutting-edge platform, SAP's most important product S/4HANA leverages in-memory computing technology provided by SAP's own HANA database, enabling real-time data processing and enhanced performance capabilities. As was the case for its immensely successful predecessor R/3, it is designed to support a multitude of business areas, streamlining operations across various domains within a single platform.

2.1.2 Core Modules of SAP

Key areas supported by S/4HANA include finance, where it offers improved financial planning and analysis; supply chain management, enhancing logistics and procurement processes; manufacturing, with optimized production planning and execution; sales and distribution, providing better customer engagement and order fulfillment; as well as human resources, where it streamlines workforce management and organizational planning. Additionally, S/4HANA integrates

advanced analytics and reporting tools, giving businesses comprehensive insights for strategic decision-making. This broad spectrum of functionalities underscores S/4HANA's role as a comprehensive, versatile, and future-ready ERP solution, adept at meeting the diverse needs of modern enterprises.

2.1.3 SAP Architecture

At its core, SAP systems rely on a robust database layer to store and manage vast amounts of data. Historically, SAP has been database-agnostic, supporting a range of databases including Oracle, Microsoft SQL Server, and IBM DB2. However, with the advent of SAP HANA, SAP introduced its own in-memory, column-oriented, relational database management system, which significantly enhances data processing speeds and analytics capabilities (Preuss 2017). With the changes in the newest iteration of SAP's ERP software, S/4HANA, being very closely linked to the properties and characteristics of the HANA database management system and its inherent benefits, S/4HANA has stopped supporting other databases, including in-memory databases of third-party providers, and can only be used in conjunction with HANA (Densborn and Finkbohner 2021).

SAP has typically been employing a three-tier client-server architecture as shown in Figure 1, consisting of the database layer, the application layer, and the presentation layer since SAP R/3 (NetLibrary, Inc 2005, 39). The presentation layer is where the end-users interact with the system, typically through SAP's GUI (Graphical User Interface) or via web browsers, typically within SAP's Fiori paradigm, in more modern implementations.

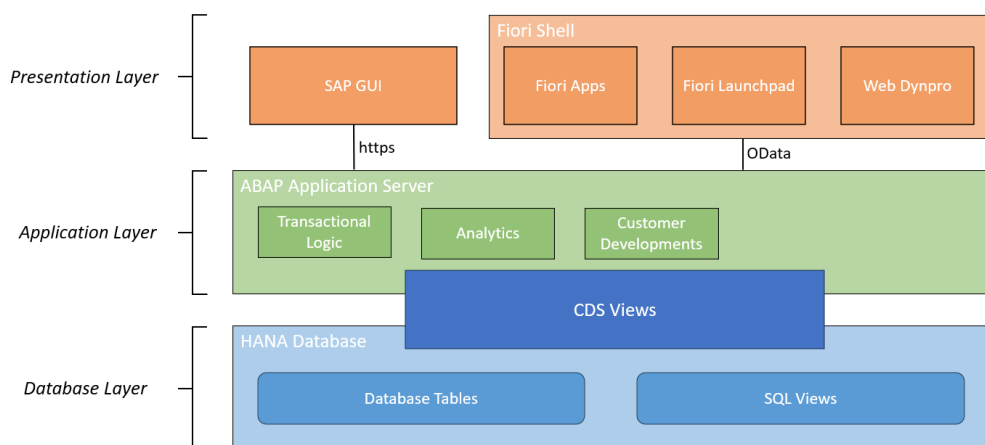


Figure 1 SAP's client server architecture

The application layer runs the actual SAP software and business logic, processing transactions and managing data exchange between the database and the clients. For this it employs SAP's proprietary and multi-paradigm programming language ABAP, which powers most of the business applications, enhancements thereof and customer-specific developments on the ERP itself.

In the context of this thesis it is important to note that the lack of standardization leading to ABAP supporting several paradigms for implementations, enhancements and basic architecture creates its own type of very deep complexity necessitating adequate documentation. This reaches from incorporating object-oriented, imperative and functional programming paradigms into one language, ever changing best practices ranging from classic ABAP to the ABAP Programming Model for SAP Fiori and most recently the ABAP RESTful Programming Model (RAP) based on the powerful CDS view technology including a so-called "Code Pushdown" (meaning moving business logic from ABAP to the database to increase efficiency) to basic extensions of the standard software which is now possible through modifications, enhancements, user exits, classical BAdIs, new BAdIs, BTEs, and others (SAP SE 2023).

SAP is designed to integrate seamlessly with a myriad of other systems and software applications, both within and outside the SAP ecosystem. This encompasses external systems like third-party analytics tools, e-commerce platforms, specific software for the core business and legacy systems but also a range modules within SAP, such as Extended Warehouse Management (EWM). Although EWM is a component of the standard installation package, it is designed to operate in a decentralized manner. Consequently, this necessitates the implementation or setup of interfaces like the Core Interface (CIF) for effective integration and functionality (Lange et al. 2020, 1089). This integration is facilitated through a range of middleware and interfacing tools, such as the SAP Integration Suite (part of the BTP) and diverse API technologies like the Business Application Programming Interface (BAPI) or Intermediate Documents (IDocs) and monitoring tools such as the Application Interface Framework (AIF), some of them proprietary.

The SAP landscape can be quite complex, often encompassing multiple instances such as development, testing, and production environments. Each environment

may have its own dedicated server and database, which requires careful management and synchronization.

Given the complexity of its architecture, integration points, and the critical role SAP plays in business operations, thorough documentation is essential. Adequate documentation ensures that the system setup, configurations, custom developments, and integrations are well recorded, facilitating maintenance, troubleshooting, and future upgrades. It also aids in training new users and IT staff, ensuring that the knowledge about the system's setup and operation is not siloed but shared across the organization.

2.1.4 SAP's Role in Business Processes

Today's companies are encumbered by siloed and unintuitive business systems with high learning curves, which complicate their processes and hinder operational efficiency. Market research firm IDC reports that such inefficiencies lead to an annual revenue loss of 20 to 30 percent for businesses (Candito 2016).

With S/4HANA, SAP offers a solution that is designed to integrate and streamline those business processes into a single, coherent system – which is exactly what companies facing those issues today are seeking (Dave, Sarkar, and G. Singh 2023, 2). This integration facilitates seamless data flow between different departments, ensuring that all areas of the business are aligned and operating on the same information. For example, billing data entered into the Sales & Distribution (SD) module can be immediately available to the finance & controlling (FICO) modules (Islam, 21), reducing the likelihood of errors.

With the incorporation of SAP HANA, businesses can now access and analyze data in real time. This capability is crucial for quick decision-making and staying ahead in a fast-paced business environment. For instance, using specific SAP Fiori apps it is possible to monitor items with reduced inventory turns and to track the shelf-life of products in real-time, facilitating quick and efficient decision-making, thus optimizing inventory levels (Densborn and Finkbohner 2021).

While allowing customers to adapt the software to their business needs via extensive customizing options, SAP also offers best practices coming from the collective best-in-class processes of SAP customers. This standardization is key to reducing process variability, improving efficiency, and ensuring compliance with

industry best practices. By using SAP's pre-configured processes, companies can benefit from the industry expertise that SAP has embedded into its software, simplifying process redesign and improvement during an SAP implementation project (Sven Denecken 2020).

2.1.5 Advances in SAP Technologies

One of the main advances technology-wise is the move from on-premise into the cloud following SAP's cloud-first strategy, which includes more recent innovations only being available to cloud customers and saw cloud services rise to 45% of SAP's revenue in Q3 of 2023 (Sayer 2024). While S/4HANA is at its core still an on-premise software with according to SAP experts up to 90 percent of its functions based on old ABAP tables from the R/3 system (Peter M. Färbinger 2023), SAP nevertheless not only provides several cloud-based deployment options for S/4HANA including a multi or single tenant public cloud edition and a private managed cloud scenario, but started following a cloud-first strategy with many innovations available strictly in the cloud and a separated code-base for cloud and on-premise (Kerkmann 2023).

Although SAP did pursue side-by-side scenarios for customer extensibility of the main software in the past, this became even more important with SAP's recent cloud strategy, feeding into a "Clean Core" approach, the aim of which is to leave the core S/4HANA instance entirely untouched software-wise by moving any custom developments onto SAP's new and heavily advertised Business Technology Platform (BTP).

2.2 Consulting

2.2.1 Role of SAP Consultants

While SAP provides numerous benefits and opportunities for continuous improvement and innovation, its implementation and ongoing management can be challenging. The complexity of the system requires skilled professionals for implementation and maintenance. Those professionals will in many if not most cases not be found inside the organization, but will have to be sourced from third-party companies in the form of SAP consultants.

SAP consultants play critical roles in both implementation and maintenance projects. Their roles and responsibilities vary depending on their area of expertise and the specific requirements of the project, although they can roughly be divided into functional and technical consultants. But while external SAP consultants are crucial for the implementation success, one common error is to rely too heavily on them while not involving company staff adequately. The consultants will leave after the project is done, while the employees have to live with the system. Hence it should be imperative to assign as many key roles and responsibilities to internal staff (Anderson 2009, 53).

Figure 2 depicts a typical project setup with key personnel in all areas on both the customer and consultant side, including a separate consultants project management role.

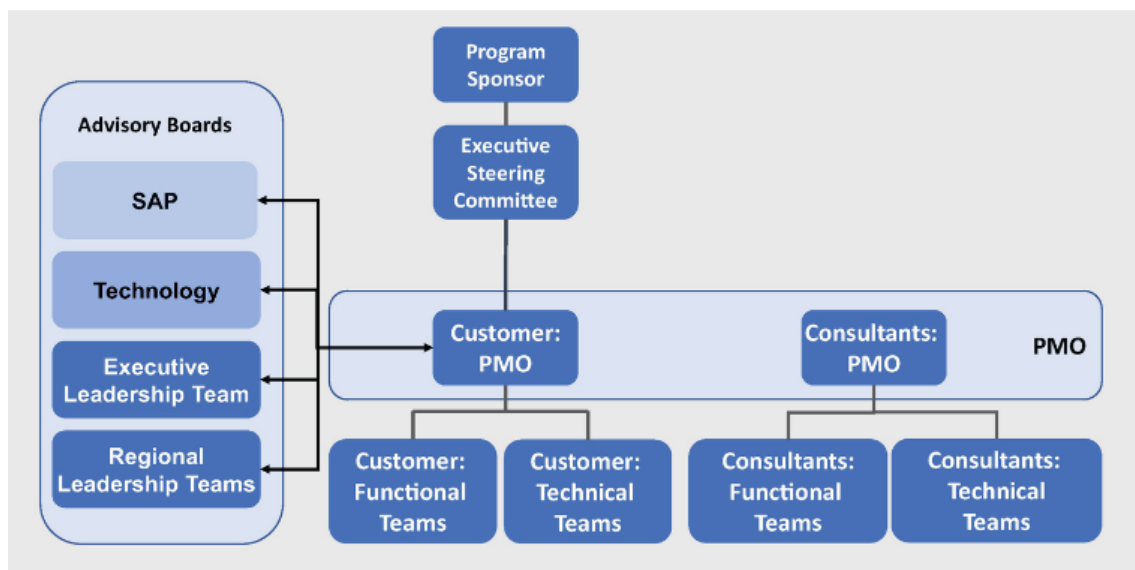


Figure 2 Typical division of responsibilities between client and consultant (Banks-Grasedyck et al. 2022)

Some of the responsibilities during implementation projects where (external) consultants are usually involved include: (Banks-Grasedyck et al. 2022, chap. 8; Anderson 2009, 625–39)

- **Requirement Analysis:** Understanding and documenting the client's business processes and requirements to determine how the SAP system should be configured and customized.
- **System Configuration & Customization:** Tailoring the SAP system to meet the specific needs of the business, including setting up modules like finance, sales, human resources, etc.
- **Custom Developments:** Implementing client-specific solutions like reports or interfaces to third-party software and leveraging the possibilities SAP offers for enhancing standard code.
- **Data Migration:** Assisting with the transfer of data from legacy systems to the new SAP system, ensuring data integrity and accuracy.
- **Testing:** Conducting various types of testing (unit, integration, user acceptance) to ensure the system works as intended.
- **Training & Support:** Creating training material and training end-users on how to use the new system.
- **Support:** Providing ongoing support after the Go-Live, especially during the Hypercare phase.
- **Project Management:** Overseeing the project to ensure it is completed on time, within budget, and to the satisfaction of the stakeholders.
- **Documentation:** Last but not least creating detailed and accessible documentation on the system's setup, configurations, developments and processes for future reference.

2.2.2 SAP Implementation Methodologies

There are two main methodologies for SAP projects designed by SAP itself. Those are Accelerated SAP (ASAP) and its successor SAP Activate. The ASAP roadmap encompasses the following six distinct phases: (V. Singh 2017, 9)

1. **Project Preparation:** Lays the groundwork for a successful project by gathering information, requirements and establishing a project organization and initial plan.

2. **Business Blueprint:** Creates a common understanding of the goals and objectives that should be met by the SAP implementation, culminating in a detailed documentation of the customers' requirements called the Business Blueprint.
3. **Realization:** Implementing or customizing and testing all processes and objectives specified in the blueprint.
4. **Final Preparation:** Completion of final testing, user training, system management and cutover activities.
5. **Go Live & Support:** Move system into a live production environment based on a cut-over plan and provide ongoing support.
6. **Run:** Ensured operability of the solution.

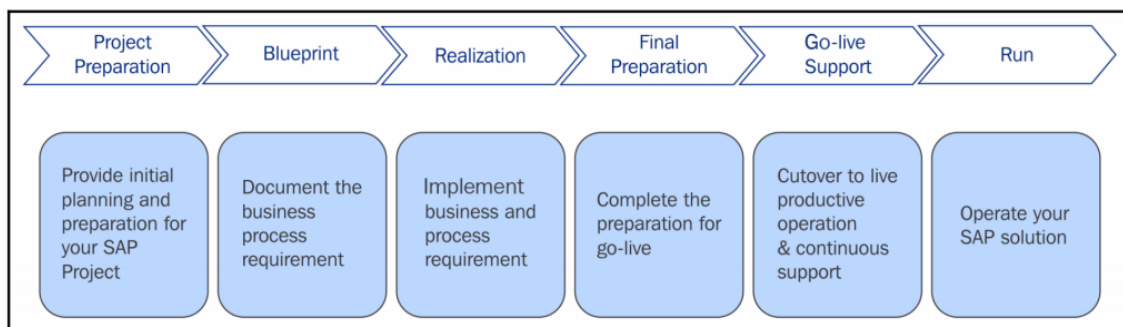


Figure 3 ASAP Implementation Roadmap (V. Singh 2017, 8)

The successor SAP Activate on the other hand comes not only with its methodology, but also SAP Best Practices and guided configuration, which can all be applied to cloud as well as on-premise implementations of primarily SAP S/4HANA, but also other products such as SAP Business Suite, SAP CRM, SAP SCM, etc. SAP Activate supports the scenarios New Implementation (data migration to SAP S/4HANA), System Conversion (technical conversion from SAP ERP to SAP S/4HANA) and Selective Data Transition (selective data migration to new S/4HANA system) and contains only four main phases: (Bardhan 2021, chap. 16)

1. **Prepare:** Foundation with the project similar to the Project Preparation phase of ASAP but bases on SAP Best Practices as far as possible to accelerate adoption and benefit from tried and tested SAP business process knowledge.

2. **Explore:** Validation of the previously created and Best-Practice-based solution and identification of differences or gaps during a fit-gap workshop.
3. **Realize:** Building and testing the final solution, similar to ASAP's Realization phase but employing the agile Activate methodology.
4. **Deploy:** Final preparation for go-live including cutover activities, end-user training and hypercare activities. This is generally where the documentation is maintained.

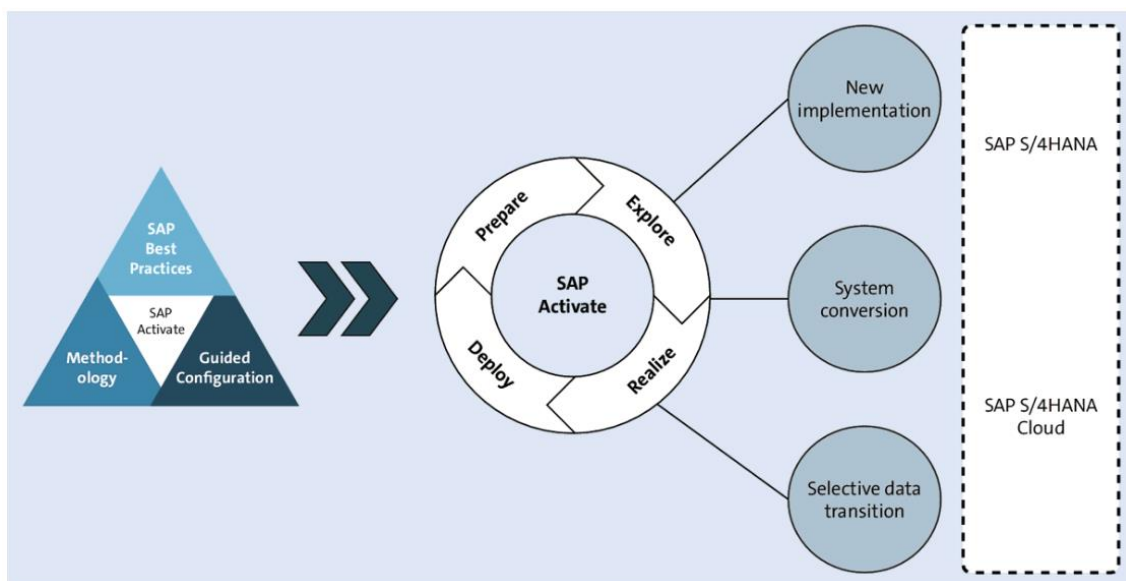


Figure 4 SAP Activate Overview and Deployment Options (Bardhan 2021, chap. 16)

In this respect it is also important to mention RISE with SAP, a very recent and comprehensive offering that facilitates the adoption of cloud-based SAP solutions which works in conjuncture with SAP Activate (Adnette Kamugisha 2021). The product was released in early 2021 and integrates various tools and services into a single package, providing customers with “business-transformation-as-a-service”. The offering includes the SAP S/4HANA Cloud Suite, various system hosting options, and the SAP Business Network Starter Pack. Additionally, it provides SAP BTP Consumption Credits and Business Process Intelligence tools. The package is beneficial for companies with limited IT resources, as it simplifies implementation and offers tailored flexibility. RISE with SAP represents SAP's shift towards a cloud-first approach, streamlining cloud adoption for their clients. (Fesko 2022)

3. Large Language Models (LLMs)

3.1 Introduction to Large Language Models

3.1.1 Definition and Evolution of LLMs

LLMs are based on language models, which are probabilistic models of a natural language capable of predicting upcoming words or assigning probabilities to sentences (Jurafsky and Martin 2023, N-gram Language Models). The first instances of these models are purely statistical models such as word n-gram language models, which assume the probability of the next word in a sentence to depend solely on the ones preceding it. (Bengio et al. 2003)

Early neural network models, which improved understanding of context and relationships, evolved into recurrent neural networks (RNNs), offering better handling of long-term dependencies in text. The adoption of neural network models in natural language processing (NLP) began around 2013 and 2014. This period marked a significant shift in the field with the introduction of three main types of neural networks: RNNs – a popular choice for dealing with dynamic input sequences common in NLP, convolutional neural networks (CNNs), and recursive neural networks. (Ruder 2018)

Basic Vanilla RNNs evolved into more advanced forms like Long Short-Term Memory networks (LSTMs), which were more effective at handling the vanishing and exploding gradient problem (Hochreiter and Schmidhuber 1997).

CNNs, a kind of feed-forward neural network that leverages convolution structure in order to extract features from data, are known for their success in computer vision and were also adapted for language processing (Z. Li et al. 2022). As opposed to RNNs, CNNs are very easily parallelizable using several GPUs for training and evaluation providing a significant acceleration in processing speed (Simonyan and Zisserman 2014). Moreover, they frequently combined with LSTMs for improved performance (Jin Wang et al. 2016; Mutegeki and D. S. Han 2020; Kim and Cho 2019).

Recursive neural networks, another approach that can be seen as a generalization of a recurrent neural network, treat sentences as hierarchical tree structures

rather than mere sequences. This approach aligns more closely with linguistic theories and allows for building representations of sequences from the bottom up. Each node in the tree computes a new representation by combining the representations of its child nodes. (Irsoy and Cardie 2014; Ruder 2018)

These developments reflect the evolution of neural network models in NLP, moving from simple, linear models to complex, hierarchical, and dynamic architectures capable of handling the nuances of human language. The real breakthrough, however, came with transformer-based models, like BERT (**B**idirectional **E**ncoder **R**epresentations from **T**ransformers) and GPT, leading to large language models (LLMs). These LLMs, including GPT-3 and its successors, represent a paradigm shift with their deep learning capabilities, vast training data, and exceptional text generation and understanding abilities. (Devlin et al. 2019)

LLMs lie within the field artificial intelligence (AI) at the crossroads of machine learning, specifically deep learning, and natural language processing. They are trained on vast amounts of text data resulting in an enormous amount of parameters in their neural networks and capable of understanding and generating human-like text, enabling them to engage in conversations, answer questions, and even create content. As such, they represent a significant advancement in the field of AI and specifically NLP. (Amaratunga 2023)

3.1.2 Components and Architecture

LLMs, in particular OpenAI's GPT-3 and higher, which is the most popular but still the tip of the iceberg of LLM-products, gained a lot of notoriety because of their ability to very quickly understand and replicate things that would be difficult or even impossible to program manually, including a broad range of linguistic subtleties, patterns and contexts.

The effectiveness of LLMs stems from the use of the transformer architecture in numerous LLMs, coupled with the extensive data they are trained on. The transformer architecture, first introduced in 2017, utilizes exclusively attention mechanisms as depicted in Figure 5, which operate akin to human focus during reading, where certain parts of the text are deemed more crucial for comprehending the overall meaning, regardless of their distance within the text.

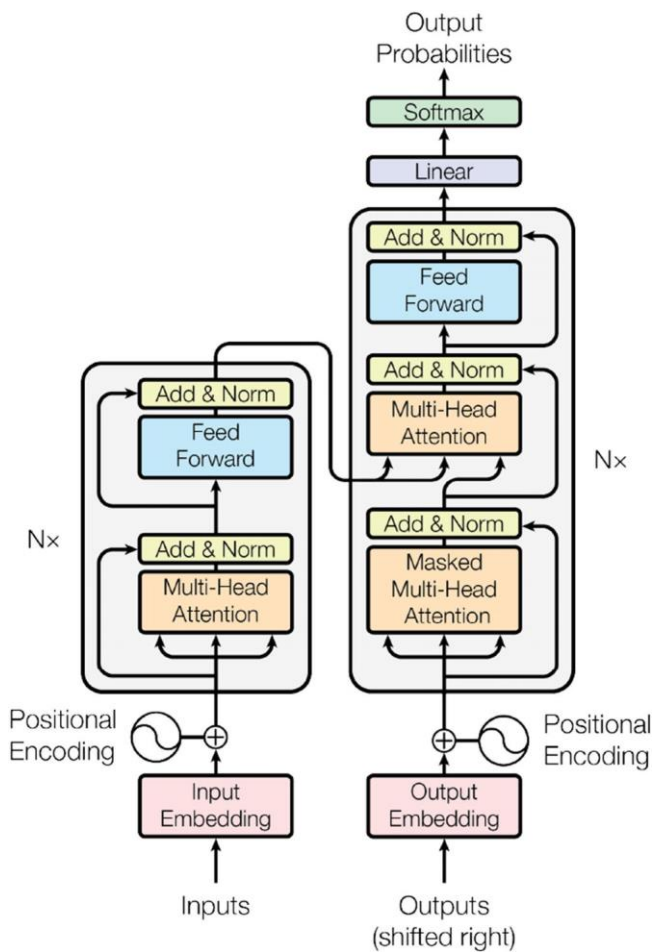


Figure 5 The transformer architecture (Amaratunga 2023)

The model is able to do this multiple times, focusing on different text segments each time and thereby examining it from multiple perspectives. While rendering recurrence and convolutions obsolete, this also enables significantly higher levels of parallelization. (Vaswani et al. 2017)

3.2 Functionality and Capabilities of LLMs

3.2.1 Language Understanding and Generation

Providing causal explanations as to how a particular LLM understands and predicts text poses a considerable challenge given that many of the most accurate and widespread models are implemented as black boxes, basing their linguistic capabilities on exploiting sophisticated yet opaque machine-learning models instead of having them programmed directly by humans. (Guidotti et al. 2019)

There are several models that are used to try to explain a given model's behavior, many of them trying to derive behavior from the body of knowledge it is based on. (Lyu, Apidianaki, and Callison-Burch 2023)

Gat et al. 2023 suggests, however, that this might not be sufficient as it is not a given that all of the knowledge and features the model consists of or is based on are consulted during its processing of a task. Instead, it may be necessary to rely on direct causality, which again poses an enormous challenge by itself. It is therefore safe to say that, considering the current state of research, many LLMs will continue to be opaque and near impossible to truly understand, fostering concerns about its safety.

3.2.2 Training an LLM

The two main steps required to train a large language model are pre-training and fine-tuning. As a third and optional step, targeted prompting techniques can be used to augment the model even further, as depicted in Figure 6. This chapter explores all the steps in more detail.

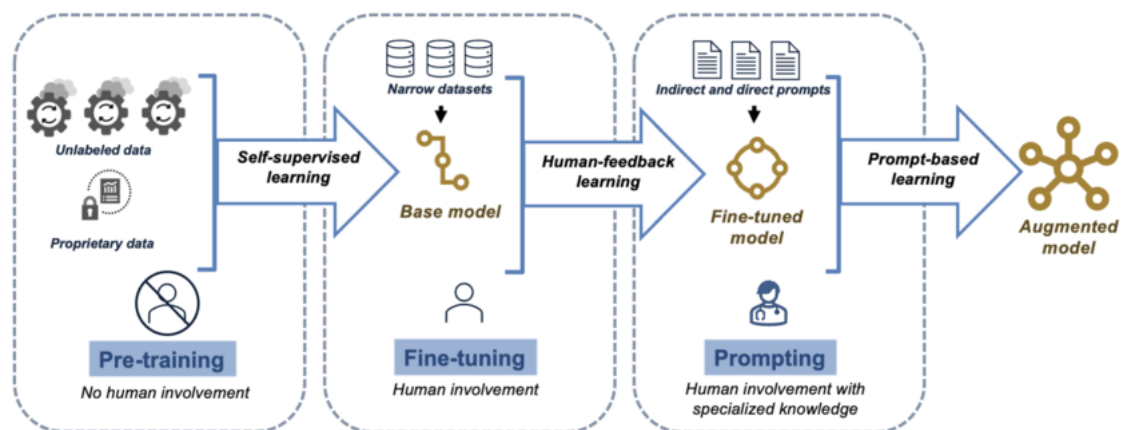


Figure 6 Overview of LLM training process (Omiye et al. 2023)

Pre-training teaches a model the basics of a language, e.g. grammar, syntax, idioms and so on. This is done based on vast, internet-scale amounts of text. This body of text is then cleaned in order to strip it of formatting issues, irrelevant information and so on. Using the resulting data, the model is then trained to predict the next words in sentences or sequences of text. A common problem is that those training materials are often kept secret as they are often times what makes the model unique. (K. Zhou et al. 2023)

Pre-trained transformer models can be divided into three distinct main categories (T. Lin et al. 2022):

- Encoder only, e.g. BERT (Devlin et al. 2019)
 - Typically used for natural language understanding tasks
- Decoder only, e.g. GPT series (T. Brown et al. 2020)
 - Delivers impressive results in natural language generation
- Encoder-Decoder, e.g. BART (Lewis et al. 2019)
 - Aims to excel at both natural language understanding and generation

Fine-tuning on the other hand prepares the model for specific tasks or applications via smaller and more specific datasets, with the key distinction between pre-training and fine-tuning being the timing of when the specific target skill is acquired. (Mao 2020)

A common method for fine-tuning a model, instruction tuning (IT), involves a supervised training utilizing (instruction, output) pairs in order to teach the model how to better follow human expectations in regards to generated text. (Shengyu Zhang et al. 2023) A notable downside to instruction tuning is the fact that this seems to only teach the model “superficial patterns, such as the output space and format” rather than have it actually understand the task better (Kung and N. Peng 2023)

It is also possible to achieve similar results on already trained and fine-tuned models through other means. One of those is few-shot prompting as proposed by T. Brown et al. 2020, which comprises giving the model specific examples of what the answer should look like before providing the actual prompt or question, as depicted in Figure 7.

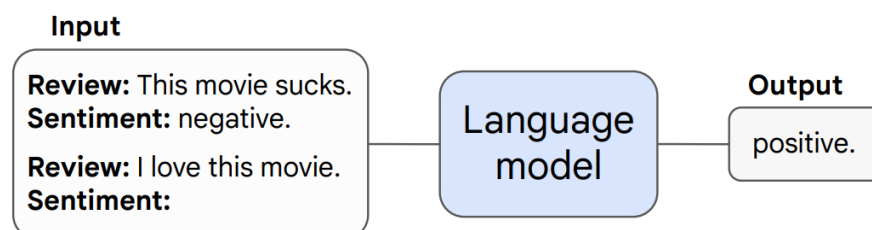


Figure 7 Example for few-shot prompting (Jason Wei et al. 2022)

Prompt engineering, a similar technique, involves a process of manually crafting input prompts to guide the LLM in generating specific types of responses. While these methods do not necessitate a potentially very cost-intensive training or fine-tuning to achieve similar results with a general purpose model, the effects are limited to but a single conversation. (Jason Wei et al. 2022; T. Brown et al. 2020; Shengyu Zhang et al. 2023))

As we know today, increasing the parameters and the body of text used for training also increases the performance – meaning that transformer-based models have the ability to scale effectively (J. Kaplan et al. 2020). Moreover the performance of general-purpose pre-trained language models (PLM) does not just increase in general with the amount of data and parameters, but is even superior after fine-tuning for a specific task to smaller models trained from scratch for the same purpose (Naveed et al. 2023).

What this means, however, is that immense resources are needed in order to create models that can compete with existing ones, leaving many researchers or practitioners at a significant disadvantage. This creates a barrier to entry for smaller institutions or independent researchers who might not have access to the same level of computational power or extensive datasets as larger organizations. While it is possible to get decent performance with modest means, i.e. in a short amount of time and using few or only one GPU (“cramming”), there are still major disadvantages (Jonas Geiping and Tom Goldstein 2023). Consequently, the development and innovation in the field of LLMs may become increasingly centralized, potentially limiting the diversity of research and perspectives in this area.

(Shengyu Zhang et al. 2023)

3.3 LLMs in Business Applications

3.3.1 Use Cases in Various Industries

LLMs ability to process and generate language in a contextually relevant manner allows for more natural and intuitive interactions with technology, marking a new era in human-computer interaction. They already offer a wide range of

applications, from customer service automation (Sadıkoğlu et al. 2023) to aiding in creative writing (Franceschelli and Musolesi 2023).

This includes areas like content creation, language translation, information extraction, question-answering systems, and chatbots. Their use is widespread in industries, such as customer service automation, educational tools, content moderation, and more.

As many models are also fluent in a diverse set of programming languages, an obvious field in which LLMs are already being productively used is software development. The advent of tools like the GitHub Copilot, offering LLM powered autocompletion of code in several coding languages (Prather et al. 2024), has brought many opportunities. This development is, however, a double-edged sword, bringing about new potential issues especially in education. (Becker et al. 2023) But the real-world application in Software Engineering does not stop there as there are also efforts to apply LLMs in testing. There are still many challenges ahead, however, like the generation of automated unit test cases. (Junjie Wang et al. 2023)

Due to their nature as natural language processing models, it follows that LLMs are primarily utilized in applications directly related to text-based tasks where natural language processing is essential and their ability to interpret and generate human-like text can be best leveraged. In the future, we can, however, expect a lot more fields in which LLMs like ChatGPT will be used as there is a near constant stream of new developments and innovations across a multitude of domains, some with seemingly few overlaps to natural language processing like dentistry (H. Huang et al. 2023) or clinical microbiology (Egli 2023).

3.4 Challenges and Limitations of LLMs

3.4.1 Technical Challenges

One significant challenge with LLMs lies in the discrepancy between the model's training goal and what users expect as they are generally trained to reduce the error in predicting words in context across extensive datasets. However, users desire the model to "follow their instructions helpfully and safely". (Shengyu Zhang et al. 2023)

Another issue is their tendency to “hallucinate”, leading them to provide false or inaccurate information while presenting them with confidence in a compelling fashion (Ji et al. 2023). This is likely to cause many users to take the hallucinations for fact, especially with the perceived competency of the models with their vast knowledge and reasoning capabilities. This may necessitate LLMs to display or allow extraction of its internal “belief” in the accuracy of a statement. (Azaria and Mitchell 2023)

Another limitation is that while LLMs do perform very well across all sorts of domains, they have problems with long-tail knowledge, that is data which is not widely available in their pre-training data.<https://proceedings.mlr.press/v202/kandpal23a.html> (Nikhil Kandpal et al. 2023)

3.4.2 Ethical and Privacy Concerns

The widespread use of tools like ChatGPT also fuel many concerns about its ethics and privacy with some even labelling it as an “infodemic threat” in fields like public health (Angelis et al. 2023) due to its ability to accidentally or even deliberately produce incorrect bodies of text that seem professional and trustworthy. This is especially dangerous in a world where misinformation is already on the rise and where there are no real measures to validate generated text as of yet.

Additionally, language models and tools building on them tend to be subject to a large array of biases that were mostly present in their training data, like gender or racial biases, or specifically implemented by their creators. (Ray 2023)

The progress in autonomous systems like self-driving cars already saw calls for their stricter regulation even before the dawn of widely used LLMs (Danks and London 2017). The same holds true for generative AI, where entirely new regulatory categories and frameworks are demanded in order to ensure that these technologies can continue to be used safely. (Meskó and Topol 2023)

Furthermore, the environmental impact of such large-scale computational requirements cannot be overlooked, as the energy demands for training and running these sophisticated models are substantial. This raises sustainability

concerns, emphasizing the need for more efficient methods and the exploration of alternative approaches that require fewer resources. (Ray 2023)

LLMs have also raised several privacy concerns due to their potential to store and reproduce sensitive information. This first and foremost includes the data used during pre-training (Ippolito et al. 2023) and fine-tuning (Miresghallah et al. 2022), often times sourced from the internet, including personal texts that might contain confidential information. The risk lies in the model inadvertently generating outputs that mirror or reveal this sensitive data, posing a threat to individual privacy. Additionally, the prompts provided during user interactions can be of a personal nature or classified, raising questions about data storage, usage, and potential misuse. (Duan et al. 2023) The challenge for regulatory bodies and also developers and companies is to establish robust guidelines and technologies to ensure that LLMs operate within ethical boundaries, prioritizing user privacy and data security. (Guidotti et al. 2019)

3.5 Popular Models

The sheer amount of new models being released each year – many of them publicly available – is staggering, as visualized in Figure 8 by W. X. Zhao et al. This chapter aims to therefore provide an overview only of the most popular current general-purpose large language models while omitting smaller or more esoteric ones and also precursor technologies including statistical language models (e.g. n-gram models), neural language models (e.g. Word2vec) and pre-trained language models (e.g. BERT and GPT1/2).

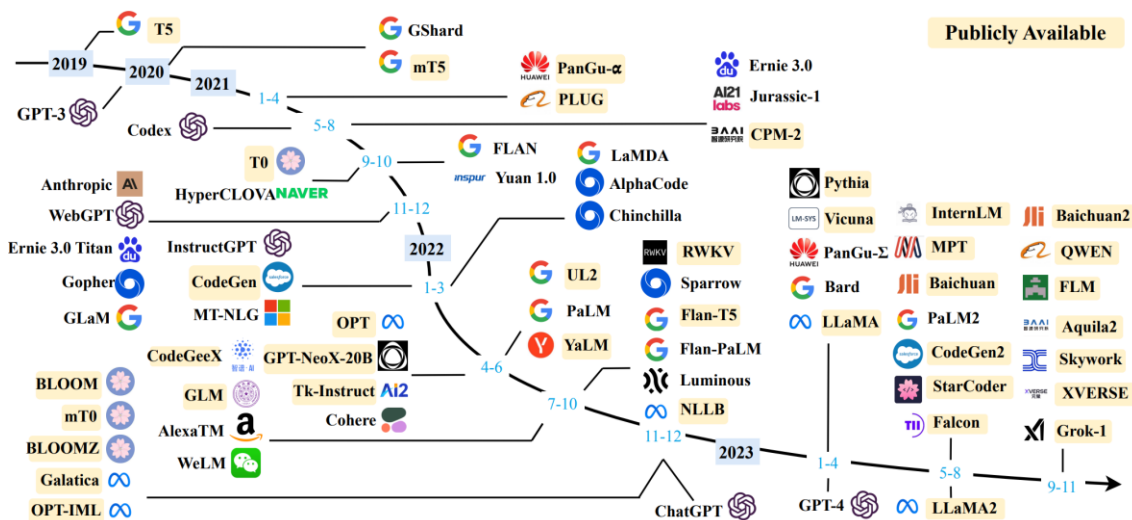


Figure 8 Timeline of recent LLMs with a size > 10B parameters (W. X. Zhao et al. 2023)

The two arguably most popular products coming to mind in the context of large language models are OpenAI's ChatGPT and Google Bard. Both are not actual language models but applications or platforms powered by underlying models to interpret and generate human-like text, offering a wide range of capabilities from answering questions to generating creative content.

For ChatGPT the LLMs used are GPT-3.5 with around 175B parameters, a subclass and tuned version of GPT-3, and GPT-4 which is only available with the premium version of ChatGPT. GPT-4s size in terms of parameters has not yet been disclosed by OpenAI but is estimated to be 1.76 trillion (Schreiner 2023).

Google Bard started out using LaMDA with up to 137B parameters (Thoppilan et al. 2022), followed by PaLM 2 with reportedly 340B parameters (Elias 2023) and is currently rolling out Gemini, the most recent LLM developed by Google featuring

an as of yet unknown number of parameters – it is expected however, that its size will exceed that of GPT-4 considerably.

Another contender, Meta Platforms (formerly Facebook), produced its own family of LLMs called LLaMA in cooperation with Microsoft, the most recent being LLaMA-2. LLaMA-2 comes in three model sizes, the largest one being 70B. (*Meta 2023*)

While comparing models is difficult and depends on a multitude of parameters, GPT-4 and PaLM 2 are generally perceived as on par. However, in a majority of scenarios GPT-4 is coming out marginally on top, demonstrating a slight edge in performance and quality (Toyama et al. 2023). LLaMA-2 on the other hand is by far the leading Open-Source LLM, offering strong metrics across the board. While being a lot faster, more efficient and cheaper to run compared to mammoths like GPT-4 and PaLM 2, it does achieve lower scores in several areas, including mathematical reasoning. (Ramlochan 2023)

4. Integrating LLMs with SAP

4.1 Integration Framework

4.1.1 Assessment of SAP System Requirements

It goes without saying that in a SAP context it is important to achieve near 100% of accuracy in the LLMs responses, as questions will often address critical processes within the business. This may require feeding the LLM with general SAP-specific as well as customer or implementation specific knowledge. Another crucial aspect is speed, as user acceptance will be very limited when faced with slow response times.

As SAP offers a substantial amount of interface technologies and platforms designed to connect to third party solutions including modern web-based ones such as SOAP or OData interfaces (Wegelin and Englbrecht 2018, 79), technically connecting the systems is less of a concern. It may, however, also be necessary to connect to knowledge bases where the organization's SAP related documentation is stored.

Moreover, factors like data security and privacy, especially in the context of LLMs using user interactions for further training. For ChatGPT there are several possibilities to ensure privacy, one of them being the possibility to disable chat history and training based on chats. (OpenAI 2023b)

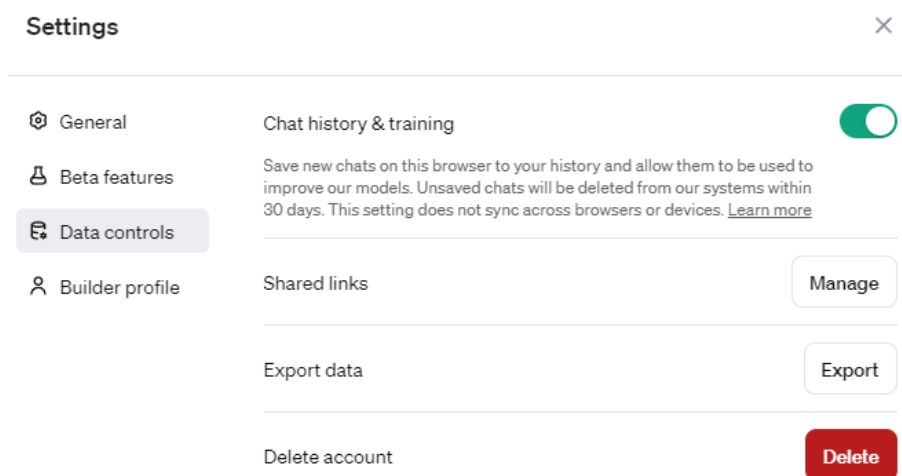


Figure 9 ChatGPT's Data Privacy Settings

This may not suffice, however, especially as OpenAI keeps records for 30 days before they are deleted, even when the chat history is disabled. Local regulations or company guidelines may require stricter measures for data management. For instance, some organizations might demand complete ownership of the data – a complex subject as illustrated by Hummel, Braun, and Dabrock (2021) – or the implementation of end-to-end encryption to protect sensitive information.

For ChatGPT or GPT-4 there are, again, solutions for this as OpenAI offers ChatGPT Enterprise addressing concerns like data ownership, encryption, compliance, Single-Sign-On and many more (OpenAI 2023a). Microsoft being a partner and investor in OpenAI also offers API access to most GPT models including GPT-4 within Azure (Boyd 2023), again addressing several potential concerns.

4.1.2 Selection of Suitable LLMs

In light of the SAP context requirements discussed previously, the selection of the most suitable option should be guided by the following criteria.

4.1.2.1 Compatibility with System Landscape

The LLM should seamlessly integrate within the existing SAP infrastructure, leveraging modern interface technologies offered by the SAP S/4HANA suite or possibly SAP BTP for cloud solutions.

Additionally, the LLM should be able to, if possible, access the company's central knowledge bases. These may include Atlassian products like Jira or Confluence, but also documentation saved in Microsoft Teams or SharePoint.

4.1.2.2 Performance

The model should offer low latency and high performance in terms of speed and accuracy, likely disqualifying smaller models on the market. It should also be scalable to handle varying loads if it is to be used in large-scale enterprise environments.

4.1.2.3 Training data

The data used for pre-training and fine-tuning by the model's creator, just like the size of the model, will have a significant impact not only on the performance, but also the effort needed to achieve the best possible answers. Specifically, it will be

very important to have SAP training and documentation data in the pre-existing knowledge of the model. ChatGPT, for example, already possesses the ability to answer basic questions relevant for SAP standard processes, as seen in Figure 10.

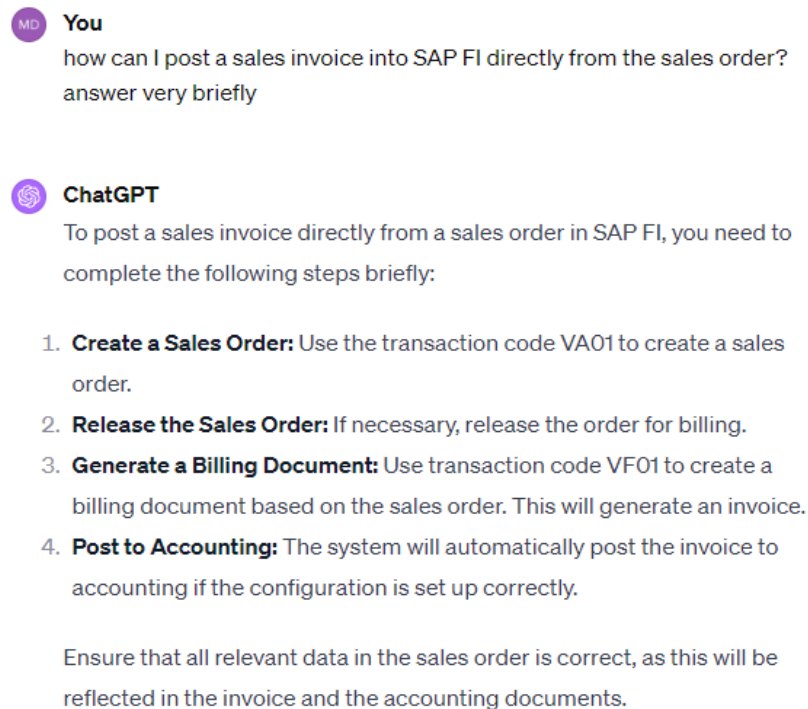


Figure 10 ChatGPT (GPT-4) answer question to FI/SD integration

4.1.2.4 Data security and privacy compliance

Given the sensitive nature of data in SAP systems, the solution must adhere to strict data security and privacy standards and regulations for the specific business sector (e.g. HIPAA) or the physical location (e.g. GDPR).

4.1.2.5 Flexibility

The ability to customize the model to suit specific business needs and processes within the SAP landscape based on specific knowledge and documentation is crucial. This includes training the model on domain-specific data to enhance its relevance and accuracy.

4.1.2.6 Cost-effectiveness

The total cost of ownership, including setup, maintenance, and operational costs, should be considered. The model should offer a good balance between cost and the value it brings to the organization.

4.1.2.7 Support and maintenance

Ongoing support and regular updates from the LLM provider are essential to ensure the model remains effective and secure over time.

4.1.2.8 Language support

The LLM should support the company's specific language or multiple languages in the case of global businesses, especially those relevant to the organization's operations.

4.1.2.9 User Experience

The LLM should enhance the user experience, offering intuitive interfaces and easy integration into existing workflows within the SAP system – if possible without the need of users signing in into a separate application.

This may necessitate the implementation of new user interfaces within the UI strategy of the organization, e.g. a new UI5 app or an ABAP report.

5. Design of Prototype

5.1 Frontend Design

This chapter outlines the design considerations for the Proof of Concept (PoC) prototype. The prototype aims to demonstrate the functionality of a Fiori/UI5 application integrated with CAP (Cloud Application Programming Model) and deployed to the SAP Business Technology Platform (BTP). This chapter discusses the visual design, architecture, and deployment strategy of the PoC.

It has been demonstrated that the user interface (UI) design of e-commerce websites plays a crucial role in fostering trust and loyalty among users. Key elements such as navigational ease, visual aesthetics, and well-structured information significantly contribute to enhancing user satisfaction, which in turn promotes trust. This relationship between UI design and user trust is critical because trust acts as a mediator that enhances the effects of a satisfying user experience on customer loyalty, particularly in the online shopping context. (Guo, W. Zhang, and Xia 2023)

Applying these insights to LLMs and chatbot applications, the principles of effective UI design can similarly have a significant impact on user trust. For LLMs and chatbots, where interaction quality and reliability are paramount, ensuring a user-friendly, aesthetically pleasing, and informative interface that fits within the context of the application could play a large role in improving user engagement and trust.

Therefore, the prototype's user interface (UI) should be developed using SAPUI5, SAP's proprietary HTML5 framework tailored for enterprise web applications (Oliver Graeff 2021). It will strictly adhere to Fiori design principles to deliver a seamless and user-friendly experience, which will in turn also make it possible to embed the prototype within a Fiori environment such as SAP's Fiori Launchpad, as depicted in Figure 11. This adherence is pivotal for the success of the prototype as the UI serves as the primary interaction point for users. Ensuring visual familiarity is crucial, as it enhances trustworthiness and fosters user confidence in the prototype and its answers.

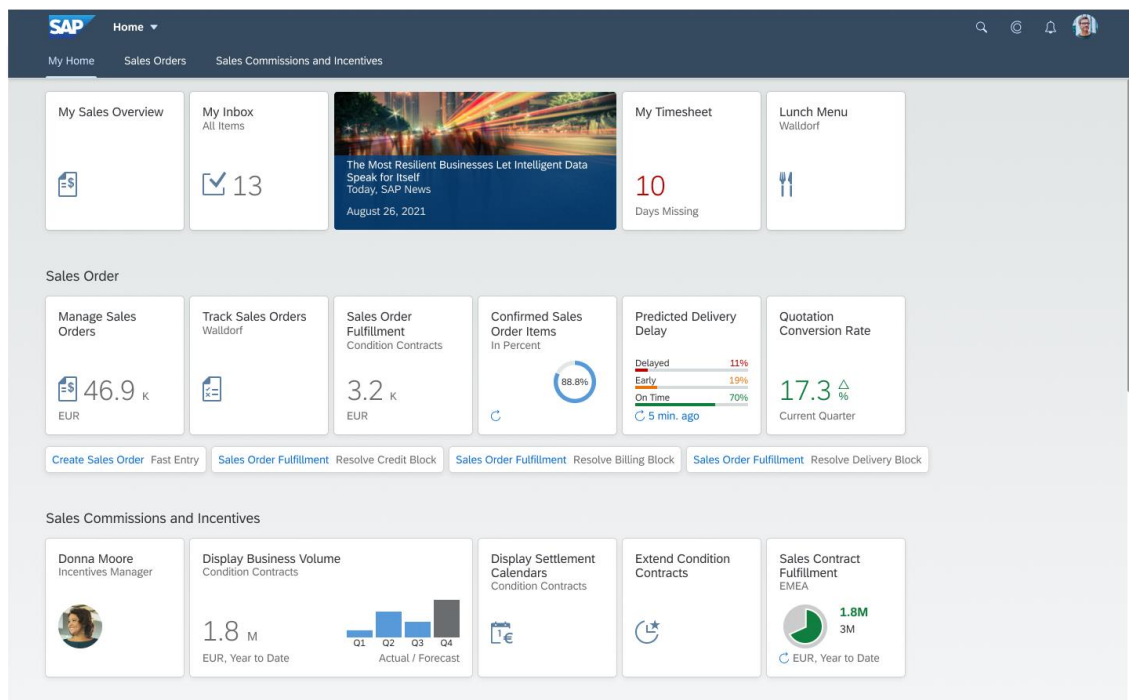


Figure 11 SAP Fiori Launchpad (SAP SE 2022)

Key design elements provided by UI5 include: (Oliver Graeff 2021)

- **Responsive Layout:** The UI will adapt seamlessly to different screen sizes, ensuring usability across various devices.
- **SAP Fiori Elements:** Leveraging Fiori elements such as tables, charts, and forms to streamline development and maintain consistency.
- **SAP Integration:** Easy integration into the SAP landscape, allowing it to access SAP systems and processes which might be necessary for more advanced applications.

5.2 Language Model

The primary workload will be managed by the language model, however. For this thesis, OpenAI's products were chosen due to their widespread adoption and reputation, as well as their powerful API and comprehensive documentation.

At the moment of writing, OpenAI provides several distinct models for integration via their API, including gpt-3.5-turbo, gpt-4, and the recently introduced gpt-4-turbo. While the "turbo" versions offer cost-effective and faster response times, gpt-4 was selected for this prototype due to its established popularity, proven

performance, larger context window, and reduced propensity for generating erroneous outputs, so called “hallucinations”. (McIntosh et al. 2024)

5.3 Test Design

5.3.1 General SAP Processes

Assessing the prototype may not be as straightforward as it appears. While this thesis explores whether LLMs can facilitate access to customer-specific SAP documentation and knowledge, the presence of material related to the SAP standard in the model's training data is also a significant part of the investigation. This pre-existing data is crucial not only for providing context to the specific customer documentation, thus enabling the model to answer questions that extend beyond the scope of customer materials, but also presents a challenge in maintaining the specificity of the responses. This arises from potential knowledge conflicts between data sources (R. Xu et al. 2024), for example, when a custom transaction developed by a customer should be preferred over a standard SAP transaction.

This underscores the importance of including an assessment of the prototype's knowledge of SAP standard processes (i.e. non-customer-specific information) in the testing phase of the prototype. Evaluating how well the model handles SAP's standard procedures and documentation is crucial. By testing both customer-specific and standard SAP knowledge, we can ensure the model's ability to accurately discern and appropriately apply the relevant information in its responses. This dual-focused assessment helps identify potential discrepancies or conflicts between customer-specific practices and SAP's standard guidelines, thereby enhancing the model's overall utility and reliability in real-world applications.

5.3.2 Customer Documentation

Upon establishing a robust baseline of the model's performance with standard data, we can then focus on its capability to deliver useful insights based on customer documentation. In order to gain a realistic and useful understanding, it is essential that the model is tested on scenarios where SAP standard processes

and transactions are in some instances replaced with customer-specific solutions, yet not universally so. This approach will help in assessing the model's flexibility and accuracy in handling real-world, dynamic and nuanced business processes.

If the model fails to prioritize specific customer data over its general training knowledge, it would ultimately prove to be useless for this purpose. Consequently, the answer to the central question of this thesis would necessarily be negative.

6. Implementation of Prototype

6.1 User Interface

The GitHub repository "cap-ui5-gpt-chat" by Mike Zschka, available under the Apache-2.0 license, provides a fully functional chat application developed using the SAP Cloud Application Programming Model (CAP) and SAPUI5. It leverages the OpenAI API for generating responses to user prompts. (Mike Zschka 2023b)

Therefore it fortunately already meets all of the prerequisites laid out during the design of the prototype in chapter 5.

Key features of the application include the ability to chat with different OpenAI models, such as GPT-4 and GPT-3.5, supporting multi-user chats with selectable "personalities" for the AI chatbot. Additionally, the application offers chat history storage via SQLite and per-chat options for streaming responses from the OpenAI API. It is also designed to be deployed seamlessly to the SAP Business Technology Platform (BTP) Cloud Foundry environment. From a technical perspective, the application features a full TypeScript implementation in both backend (SAP CAP) and frontend (SAPUI5), utilizing advanced TypeScript features like decorators and dependency injection in the backend. The chat UI is built using SAPUI5, incorporating custom list item controls and the integration of external modules. (Mike Zschka 2023a)

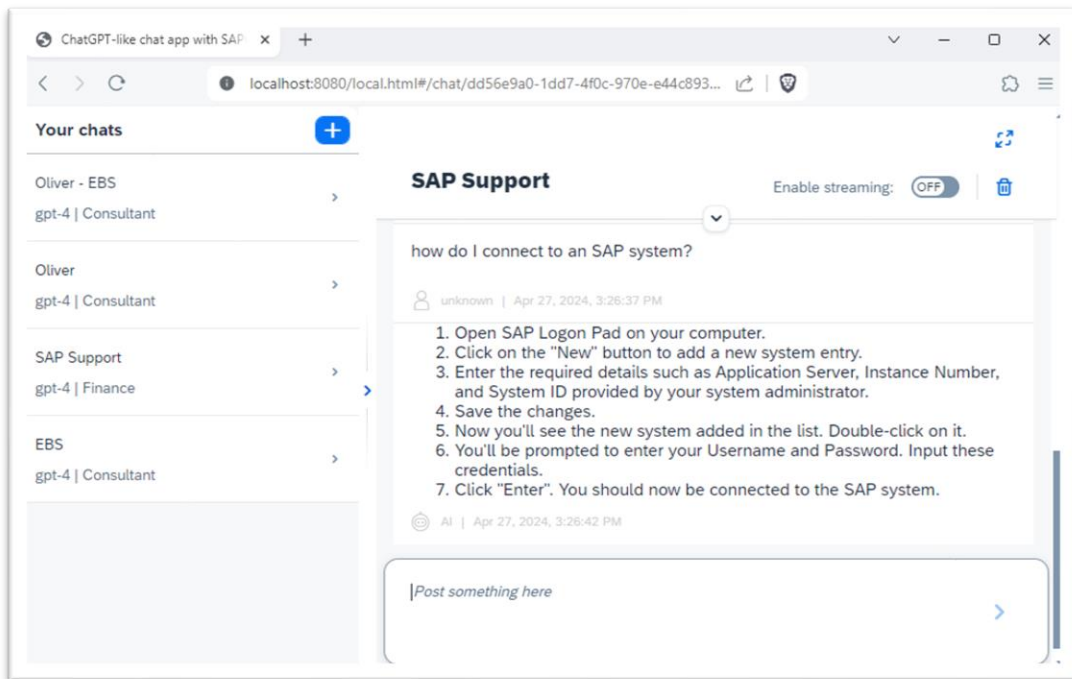


Figure 12 Screenshot of the Locally Deployed Application

The selectable “personalities”, stored in a CSV file and deployed to the SQLite database, were adjusted to better align with the thesis objectives. Presented below are two selectable personalities along with their corresponding instructions, guiding the model to tailor its responses accordingly:

- **Finance:** AI describes things short, simple and concise for an employee in the finance department, focusing on processes, workflows and requirements of the system
- **Consultant:** AI describes things short and concise from a consultants perspective, focusing on the big picture and mentioning (customizing) transactions

6.2 Training Material

As discussed in detail in chapter 3.2.2 of this thesis, there are three main procedures of training an LLM. Those include pre-training, fine-tuning and prompt engineering.

With pre-training being too time-consuming and likely cost-prohibitive for an application providing access to an ever increasing number of documents, that may also be updated regularly, it falls flat for usage in this thesis. Fine-tuning on the other hand would be a sub-optimal solution to begin with, as it is intended for training the model for very specific tasks or answer styles and “is less reliable for factual recall” (Ted Sanders and Mike Heaton 2022). This is not to say that pre-training and fine-tuning could not have their merits in enhancing the model used for an application like this – they would play a minor part, however, compared to the data provided as part of the prompt.

The downside with providing the entire context within the prompt is the limited context window, especially with GPT-4, which only includes up to 8,192 tokens (OpenAI 2024a). This is a limit that is easily reached with larger documentations and longer interactions with several follow-up questions.

In scenarios where there is a plethora of documents for the model to potentially draw upon, a solution is necessary to narrow down the selection to only those that are directly relevant for answering the user's query. One straightforward approach could involve the user specifying the context of the question beforehand, although this may not always be feasible as the user may be lacking the required context.

A powerful method to solve this dilemma is implementing a Search-Ask procedure as described by Ted Sanders and Mike Heaton 2022. Initially, the process involves searching a text library for sections relevant to a given query. This is followed by the 'Ask' step, where the relevant text is inputted alongside the query to GPT, which then generates an answer. This method avoids the pitfalls of fine-tuning by using a dynamic 'open-book' approach, ensuring the AI utilizes the most relevant and up-to-date information for each query.

This is accomplished by leveraging embeddings, which are high-dimensional vectors representing the semantics of the required texts – in our case documentations. These so-called embeddings are then persisted in a database. Each user query is then also used as basis for an embedding. The query embedding is then used to rank the persisted embeddings based on their relevance to each other, allowing the system to retrieve the text(s) of the most relevant

embedding(s). The result is then provided as one prompt with the question to the model.

For this particular prototype, given the evaluation of only one scenario or documentation file, it was not necessary to implement this complex Search-Ask logic in order to answer the research question. Instead, a straightforward approach was chosen where the documentation is included in the initial prompt with preceding instructions on how to formulate responses. This approach should yield results comparable to those achieved with embeddings, if not identical.

```
Use the below documentation to answer the subsequent
question. If the answer cannot be found, give an answer
based on your general knowledge but start your answer
with: "I could not find company specific information on
this question, but generally"
```

```
Article:
```

```
"""
```

```
<documentation>
```

```
"""
```

The documentation itself is attached as PDF to this document in Appendix 11.2. For use in the prompt, the original Word file was first saved as plain text.

6.3 Operational Costs

The application also has to be set up with an OpenAI API key in order to function properly. This requires an OpenAI account with sufficient funds, as there is a price tag on each request based on the amount of tokens used in the transaction. According to (OpenAI 2024b), a token “can be thought of as pieces of words”.

Figure 13, created based on usage data taken from OpenAI’s API platform, showcases the funds spent per interaction in the context of the interviews conducted for this thesis, each including two separate chats – one about general SAP topics, and one regarding the specific scenario based on extensive custom

specification, which had to be submitted as part of the prompt, therefore drastically increasing the amount of tokens required.

This demonstrates that operating an application such as the prototype inevitably entails some substantial financial investment, especially with more extensive documentation and a large and active user base. However, considering the potential savings in terms of time and possibly reducing the need for support staff, a strong case may still be made for its economic viability. This, however, is not part of the investigation for this particular thesis.

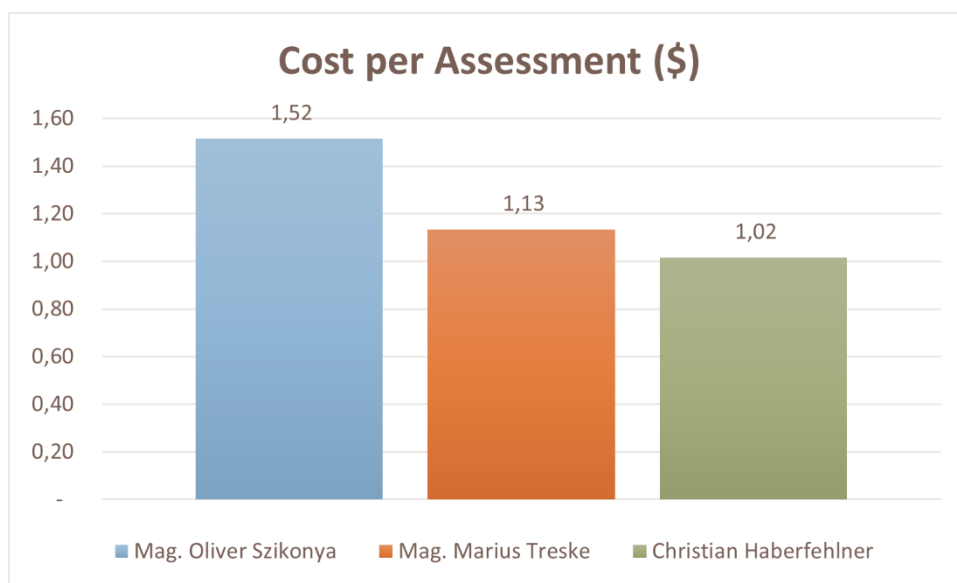


Figure 13 Visualization of OpenAI Usage Data Extract

7. Evaluation

7.1 Justification of the Selected Method

To answer the research question and provide a comprehensive assessment of the prototype for the GPT-4 based chatbot tailored for SAP-related queries (customizing, transactions, developments, etc.), the qualitative content analysis according to Mayring (2000) was chosen as the primary methodology. This approach allows for an in-depth exploration of the practical application and effectiveness of the chatbot within company-specific contexts by proven experts by utilizing qualitative, guided interviews.

In this study, qualitative interviews will be conducted with key stakeholders who possess specialized knowledge in SAP implementation, support and rollout projects. These stakeholders will include individuals directly involved in SAP project management, customization, development, (key) user training, documentation and other relevant areas.

An important selection criterium for interview participants is firsthand experience in utilizing the chatbot prototype within their SAP-related workflows. To this end the experts will be provided with access to the prototype to explore its capabilities freely before the interview commences. They will be encouraged to pose typical questions from their fields of expertise and seek meaningful responses. After this initial exploration phase, a specific scenario will be introduced to evaluate the model's ability to address inquiries regarding a heavily customized process. This process, more thoroughly outlined in chapter 7.2, was developed for a specific client and operates in parallel with standard workflows and sometimes even supersedes them.

The goal of these interviews will be to gain insights into the usability, functionality, and overall impact of the chatbot prototype in addressing company-specific SAP challenges. Additionally, the interviews will explore how the chatbot integrates into existing workflows, the effectiveness of its responses, and any challenges encountered during its implementation and usage. Another aspect of the evaluation comprises the assessment of the trust those experts have into the specific prototype and the technology in general.

Interviews will be conducted either in person or through online video conferencing platforms such as MS Teams, ensuring flexibility and accessibility for participants. All interviews will be recorded with participant consent, and confidentiality regarding sensitive company information will be strictly maintained.

By employing qualitative interviews, this study aims to provide a nuanced understanding of how chatbots based on LLMs like GPT-4 can effectively support SAP-related processes within diverse organizational contexts. The findings from these interviews will contribute valuable insights for refining and optimizing the chatbot's functionalities to better meet the specific needs of companies utilizing SAP systems.

7.2 Scenario for Assessment

The scenario originates from a real-world implementation involving a European creditor protection organization. This organization manages various tasks such as insolvency proceedings and debt collection. Consequently, the company must handle a diverse range of incoming payments from debtors, debt collectors, lawyers, and others, each pertaining to distinct business cases. To manage this complexity, a sophisticated framework for processing bank statement data was implemented. This framework operates alongside the standard SAP process and often entirely bypasses them entirely.

As such, this should pose a considerable challenge to the model, as it will be tasked with navigating between two distinct and mutually exclusive parts of its knowledge: the SAP-specific data which was part of the material the model has been trained on by OpenAI, and the customer specific documentation provided to it as part of the prototype. The results of the evaluation of this particular scenario are central for answering the core question of this thesis.

The customer documentation utilized for this study was originally written in German and has been translated into English. Additionally, all references to the original company were removed, with the company now being referred to as "ABC." The resulting document was then converted into a plain text format, which resulted in the loss of all images and attachments. The quality of the

documentation is relatively low, reflecting its different authors and incremental development without a unifying concept. This characteristic makes it an ideal candidate for evaluation, as it exemplifies how documentation is frequently compiled under time constraints and without systematic quality assurance in real-world scenarios. Apart from general descriptions, it also contains documentation of customizing, custom ABAP developments and references to additional documents, providing further potential interesting data points in the evaluation.

7.3 Methodology: Qualitative Content Analysis

Qualitative data collection methods, such as guided interviews, produce verbal material that researchers transform into data through analytical processes. Acquiring relevant knowledge is achieved by interpreting the interview transcripts. (Strübing 2018, 4) This is generally achieved without the use of strict measuring. (Bortz and Döring 2006)

The interviews are conducted using a pre-established guide (refer to attachments, 11.1) and are subsequently transcribed to serve as foundational material for qualitative analysis. To evaluate these transcripts, (Mayring 2015)) method of Structuring Content Analysis is employed, the aim of which is to evaluate the interviews based on the criteria necessary to assess the prototype.

While the interviewees possess expertise in their respective fields, particularly in SAP implementation projects encompassing documentation, trainings, and knowledge transfers, they demonstrate limited familiarity with LLMs, with some having no experience using them whatsoever. Given the challenge of finding experts well-versed in SAP, documentation/trainings, and LLM usage, it was decided to adhere to a more structured, guided interview format rather than opting for the expert interview subtype within guided interviews, as expert interviews would necessitate a deeper knowledge and understanding of the material (Strübing 2018, 106 ff.) for which the mere assessment of a prototype is not sufficient.

During this process, selected text excerpts from the transcripts are marked according to predefined categories (see chapter 7.3.4), and then coded, summarized, and refined. These marked excerpts are categorized in an evaluation

table. This extensive and categorized data will be utilized in chapter 7.4 of this work to provide important insights.

The findings from this empirical investigation should allow for a comprehensive conclusion to the research question. However, given the limited number of interviews and the broad scope of diverse areas and challenges in the world of SAP implementations and documentation, but also the possible pitfalls of LLMs, it is impossible to explore every aspect. Instead, this thesis aims to compare empirical data gained based on the very specific aforementioned scenario and the experts' assessment of which prompts are realistic and important, and draw conclusions from that.

7.3.1 Introduction of Interview Partners

7.3.1.1 Mag. Oliver Szikonya (adesso orange Austria)

Oliver is a seasoned SAP professional with almost three decades of SAP experience, dating back to 1996. As a lead consultant, he specializes in Sales and Distribution (SD) with additional expertise in Customer Credit Management (CCM) and logistics in general. Apart from that he could also gain some knowledge in SAP related customer developments, as he is responsible for coordinating development efforts in his firm. Oliver's expertise extends to conducting user trainings and creating comprehensive documentation to support users in navigating SAP systems effectively. While he has explored LLMs and ChatGPT out of curiosity, he does not regularly integrate them into his professional practice.

Oliver's wealth of experience in SAP implementation projects, coupled with his proficiency in user training and development, positions him as a valuable interview partner. His curiosity about emerging technologies highlights his adaptability and willingness to explore innovative solutions, offering insights into potential applications within the SAP ecosystem.

7.3.1.2 Mag. Marius Treske (self-employed)

Marius is a SAP consultant and project manager with thirteen years of SAP experience. His career began as an SAP FI (Financial Accounting) consultant. Over time, he expanded his expertise to include SAP CO (Controlling). For the past eight years, Marius has excelled in the role of a project manager, where he has effectively guided teams through complex SAP projects.

Marius's approach to SAP project management is notable for incorporating modern tools such as ChatGPT. He has successfully utilized this particular tool to facilitate brainstorming sessions and create detailed migration test concepts, enhancing the efficiency and effectiveness of project execution. This experience underscores his openness to leveraging new technologies to solve traditional problems, a quality that distinguishes him in the field.

Additionally, Marius has a proven track record of managing projects from inception through to completion, ensuring alignment with client objectives and seamless integration of new SAP modules. His comprehensive understanding of SAP functionalities, combined with his project management skills, makes him a great fit as an interview partner for the question of the possible role of LLMs in the world of SAP documentation.

7.3.1.3 Christian Haberefehlner (adesso orange Austria)

Christian is a Senior Financial Accounting (FI) consultant and Team Lead with nearly two decades worth of experience in SAP projects. Starting out as an inhouse consultant and later changing to the external side, his expertise spans from project inception to execution, with a strong focus on documentation and user training. Over the years, Christian has been instrumental in guiding teams through the complexities of SAP FI in SAP implementations, emphasizing comprehensive documentation practices and effective training programs to enhance (key) user competency.

Although he has not yet engaged with Large Language Models (LLMs) in his professional practices, his deep understanding of SAP FI functionalities, including detailed knowledge of the specific scenario outlined in chapter 7.2, and project leadership makes him a valuable interview partner. Christian's extensive background should provide insightful perspectives on how traditional SAP environments might integrate or benefit from emerging technologies like LLMs.

7.3.2 Interview Guide

The interview guide (refer to Appendix 11.1) was provided to all interview partners in advance, offering them the opportunity to prepare adequately. Throughout the interviews, some of the questions outlined in the guide were adjusted or reorganized as needed to ensure a smooth flow of conversation.

The aim of the interview guide is to provide a coherent structure to the interview while remaining as minimally restrictive as necessary, allowing for the exploration of significant contexts beyond the formulated questions. (Strübing 2018, 102 ff.)

7.3.3 Process of Transcription

The interviews were transcribed and evaluated following the transcription rules outlined below:

Rule	Description
Transcription Accuracy	Each interview is transcribed verbatim. However, to maintain clarity, filler words such as "um," "uh," and any repetitive words are omitted from the transcript.
Numerical Representation	All numbers are represented in their numerical form within the transcript.
Speaker Identification	Contributions from the interviewer are prefaced with "Q:" Responses from interviewees start with "A:".
Confidentiality	Personal information collected from interview participants is not recorded in the transcript. Such data are only briefly described in the thesis itself, and only with the explicit consent of the interview participants.

Interview transcriptions adhere to the following process:

- 1) The interviews are recorded using the standard voice recording app on an iPhone.
- 2) The audio file is then transcribed with the help of **Sonix** (<https://sonix.ai/>).
- 3) The transcript is re-checked by listening to the original recording.
- 4) Finally, the transcript undergoes proofreading to correct any spelling errors or ambiguities.

7.3.4 Evaluation Categories

The categories depicted in

Code	Description	Occurrences
EC1	Usefulness of the model	10
EC1.1	Usefulness for consultants in particular	3
EC1.2	Usefulness for (key) users in particular	3
EC2	Answer accuracy	4
EC3	Incorrect answers and model hallucination	3
EC4	Comparison to static legacy documentation	3
EC4.1	Advantages over static documentation	4
EC4.2	Disadvantages over static documentation	6
EC4.3	Necessary skills to utilize LLM	2
EC5	Answer quality and style	4
EC6	Feasibility for productive application	3

Figure 14 or codes have been employed for the evaluation of the interview transcripts.

Code	Description	Occurrences
EC1	Usefulness of the model	10
EC1.1	Usefulness for consultants in particular	3
EC1.2	Usefulness for (key) users in particular	3
EC2	Answer accuracy	4
EC3	Incorrect answers and model hallucination	3
EC4	Comparison to static legacy documentation	3
EC4.1	Advantages over static documentation	4
EC4.2	Disadvantages over static documentation	6
EC4.3	Necessary skills to utilize LLM	2
EC5	Answer quality and style	4
EC6	Feasibility for productive application	3

Figure 14 Evaluation Codes and Number of Occurrences

7.4 Results

This chapter aims to provide a comprehensive analysis of the results, offering a balanced view that considers both the strengths and the areas for improvement observed in the language model, showcased using actual examples of the expert's interaction with and assessment of the prototype. Through this detailed exploration, we aim to contribute valuable insights into the development and refinement of such language models for application in highly specific SAP implementations in order to help end users with learning the ropes and achieving their goals using the systems, and consultants with getting a better understanding of a particular systems peculiarities.

7.4.1 Quality and Style of Responses

The responses generated by the LLM have demonstrated both high quality and a favorable style. Despite the complexities inherent in the SAP landscape, the vocabulary used by the model remains as clear and accessible as possible. (interview with Mag. Oliver Szikonya, 2024, April 26)

Additionally, a particularly positive aspect noted was GPT-4's propensity to offer solutions in a well-structured and step-by-step format as demonstrated in Figure 15 (interview with Mag. Oliver Szikonya, 2024, April 26, 27–35; interview with Mag. Marius Treske, 2024, April 29, 50–54), which enhances the ease of understanding and following the provided instructions. This structured approach to problem-solving has been well-received, reflecting the model's utility in navigating the intricate details of SAP systems effectively.

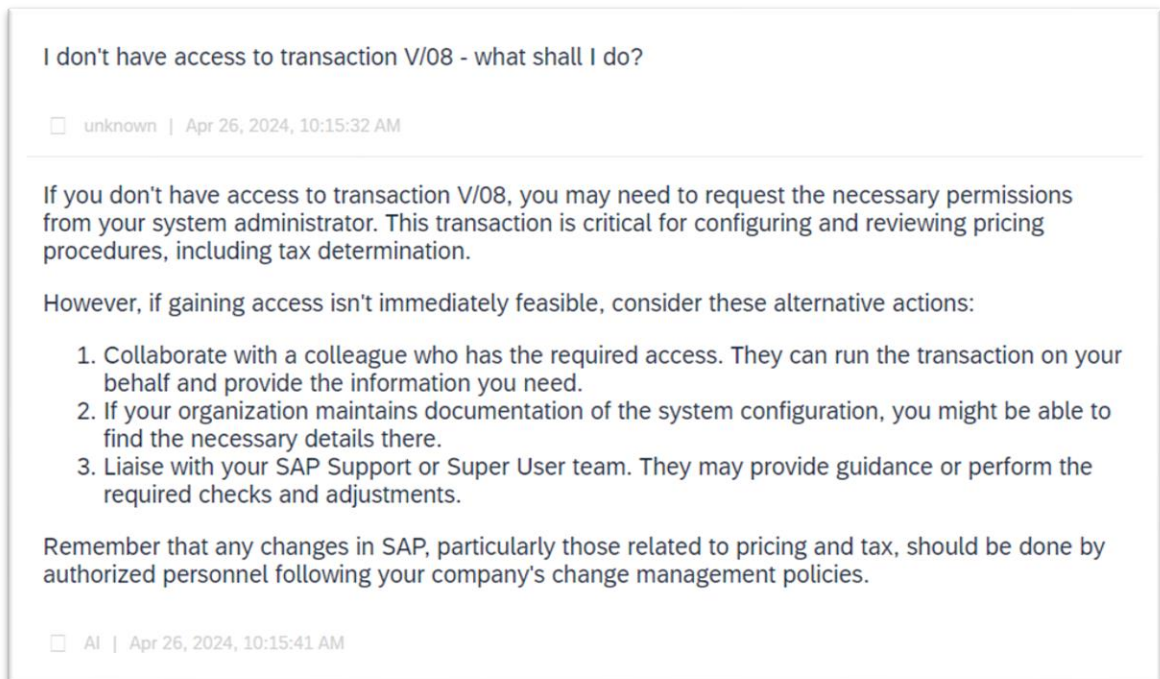


Figure 15 Helpful, factually correct and well-structured response

At one point the answers were even said to feel like SAP standard documentation within the system itself (interview with Christian Haberfehlner, 2024, May 2, 33–35), which might not always be an upside, however (interview with Mag. Marius Treske, 2024, April 29, 27).

While there were occasional errors in the responses it generated, these inaccuracies were typically easily resolved through follow-up questioning, which effectively led to the correct information, provided that the information in question was part of the training or context provided to the model. (interview with Mag. Oliver Szikonya, 2024, April 26, 8-12,83-86)

When dealing with topics the model does not have deeper knowledge on, the model can in many cases still provide a solid overview and useful pointers. However, when pressed for detailed information, it may start producing inaccurate or unhelpful data. (interview with Mag. Marius Treske, 2024, April 29) A prime example of this fact can be found in Figure 16, where the model conveys a helpful and completely accurate big picture, but when asked about details goes on to provide transaction codes that are at best marginally relevant and in all

probability already correctly maintained (e.g. translation ratios or accounts for exchange rate differences).

How to handle hyperinflation accounting in Turkey during a SAP implementation?

unknown | Apr 29, 2024, 12:26:43 PM

Handling hyperinflationary accounting in countries like Turkey during a SAP implementation requires a strategic approach.

Firstly, you would need to configure parallel currencies in the SAP system. This involves setting up both a local currency (TRY: Turkish Lira) and a hard currency (usually USD or EUR), which is stable and not subject to hyperinflation. You can then use the hard currency for reporting, which provides a more accurate representation of the business's financial health.

Secondly, you must implement index-based accounting, where you adjust the accounts to reflect the changing value of the currency. SAP's parallel valuation functionality allows this by enabling you to maintain multiple parallel ledgers.

In addition, General Ledger (GL) accounts must be correctly classified as either monetary or non-monetary. Monetary accounts (like cash, payables, or receivables) are not adjusted, while non-monetary accounts (like inventory, property, plant, or equipment) are.

Lastly, you must set up specific transaction types for revaluation, which will allow you to adjust these non-monetary accounts periodically.

It's worth mentioning that all these steps must comply with International Accounting Standard (IAS) 29, and it's advisable to work closely with the client's finance team to ensure accurate and compliant financial reporting.

AI | Apr 29, 2024, 12:26:56 PM

We decided to use EUR as hard currency (second functional currency). All prerequisites which you mentioned are finalized. Which transaction codes do I have to use to enter the settings in the system?

unknown | Apr 29, 2024, 12:29:03 PM

Great, now that you've completed the prerequisites and decided to use EUR as your hard currency, you can proceed with the following steps. Please keep in mind that these transactions should be executed by a qualified SAP Consultant or Super User. Remember to test thoroughly in your development or quality system before making any changes in production.

1. **Define Currencies of Leading Ledger (OB22):** Here, assign your hard currency (EUR) to your leading ledger as the second currency.
Note: Your first local currency would be TRY, and your company code currency may also be TRY if it's not already set to EUR.
2. **Define Translation Ratios for Currency Translation (OBBS):** This is where you define the exchange rate ratio between your local currency and your hard currency.
3. **Define Accounts for Exchange Rate Differences (OB09):** Here, you need to specify GL accounts for exchange rate differences.
4. **Prepare Automatic Posting for Foreign Currency Valuation (OBA1):** Here, set up automatic postings for foreign currency valuation.

Remember, this only scratches the surface of configuration and doesn't cover all the possible scenarios one might encounter during such an implementation. For example, additional configuration might be needed based on specific business needs or reporting requirements. Always consult with an expert or refer to official SAP documentation for more detailed guidance.

AI | Apr 29, 2024, 12:29:18 PM

Figure 16 Accurate in providing the big picture, but unhelpful with the details.

7.4.2 Comparison to Static Documentation

The primary advantage over static documentation highlighted is the ability to ask questions ranging from general to specific, which guides the user effectively through the process of finding relevant information. This is contrasted with traditional methods such as sifting through lengthy documents without sufficient guidance. (interview with Mag. Oliver Szikonya, 2024, April 26, 54–60; interview with Mag. Marius Treske, 2024, April 29, 57–59)

Additionally, the LLM is able to provide helpful and intelligible information even with low-quality and overly complex source material, that would be hard to read on its own. (interview with Mag. Marius Treske, 2024, April 29, 86–90)

A key drawback noted is the need to formulate questions accurately, which can be challenging, especially for those new to a subject. Without a foundational understanding or the correct vocabulary, users may struggle to ask the right questions or comprehend the larger context. (interview with Mag. Oliver Szikonya, 2024, April 26, 60–68; interview with Christian Haberfehlner, 2024, May 2, 83–86; interview with Mag. Marius Treske, 2024, April 29, 59–62)

A comparison to conventional search methods was also drawn, pointing out that both rely on identifying correct keywords and navigating through data efficiently. However, language models like ChatGPT can offer detailed responses and adapt to user needs based on the specificity and clarity of the questions posed. (interview with Mag. Oliver Szikonya, 2024, April 26, 71–76)

Additionally, the benefits of detailed documentation were established, such as the ability to include images and screenshots, which might be more intuitive for some users compared to the (current) text-only responses from a language model. (interview with Christian Haberfehlner, 2024, May 2, 68–71) There is also an emphasis on the potential for inexperienced users to miss important detail due to not knowing enough about the topic in order to know which additional questions to ask. When reading through a thorough documentation, the important parts on the other hand will likely be hard to miss. (interview with Christian Haberfehlner, 2024, May 2, 75–76)

7.4.3 Usefulness

While we established the difficulty for users with little knowledge of the topic in question to formulate good questions, thereby limiting the usefulness of the model in these cases, it was noticed that in these cases the model may still provide a very helpful and comprehensible general overview of a given topic which may act as a starting point in a further investigation. This was especially true for its general knowledge which is not based on specific documentation provided to the model. (interview with Mag. Marius Treske, 2024, April 29, 24–26)

Similarly, it provided a holistic approach to problems, providing alternative options and solutions on how to tackle an issue, which again fosters comprehension for newcomers to a topic. (interview with Mag. Oliver Szikonya, 2024, April 26, 18–21)

For consultants in particular, LLMs also have the ability to be highly useful, as they are able to provide not only a business view many consultants are lacking, but also in some cases detailed technical answers regarding customizing and ABAP development. (interview with Mag. Oliver Szikonya, 2024, April 26, 45–51; interview with Christian Haberfehlner, 2024, May 2, 41–42)

Key users may also benefit, provided the information isn't overwhelming. It can guide them on whom to consult for specific issues. For instance, if there's a missing authorization, they should contact their system administrator as shown in Figure 17. Similarly, for changes to complex material master data properties, they should speak with their in-house consultant. (interview with Mag. Oliver Szikonya, 2024, April 26, 38–45)

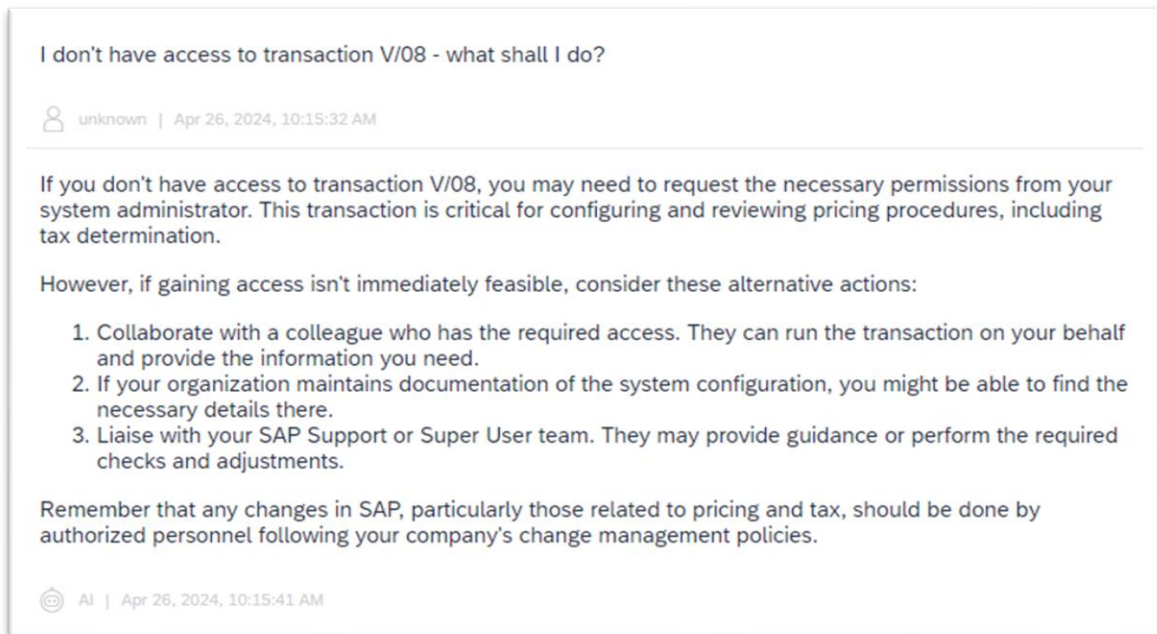


Figure 17 LLM Suggesting to Contact System Administrator

Newcomers to a company may also utilize an LLM to better understand the documentation and onboarding files presented to them. (interview with Mag. Marius Treske, 2024, April 29, 84–86)

In one of the interviews, an interesting point was also made regarding managers, who were not explicitly thought to be of interest for this particular study as the assumption by the author was that mainly (key) users and consultants would be interested in information on SAP processes and documents. However, managers may be required to make judgements and decisions in SAP projects which demand a certain, if superficial, understanding. Providing this kind of general, broad overview is what an LLM is particularly well-suited to do (interview with Mag. Marius Treske, 2024, April 29, 41–47). This also holds true for many high-level tasks in general, e.g. process definitions and architectural work (interview with Christian Haberfehlner, 2024, May 2, 42–48).

7.4.4 Challenges and Opportunities

7.4.4.1 Context

To effectively utilize the LLM in the prototype, it proved crucial to provide specific context with each prompt. For instance, although the model was equipped with a

predefined "personality" as detailed in Chapter 6.1, this personality only guided the stylistic aspects and focus of responses and did not specify that responses should be tailored to SAP-related topics.

This limitation became evident when the model delivered responses in a general tone that required contextual adjustments to align with SAP-specific content, as demonstrated in Figure 18.

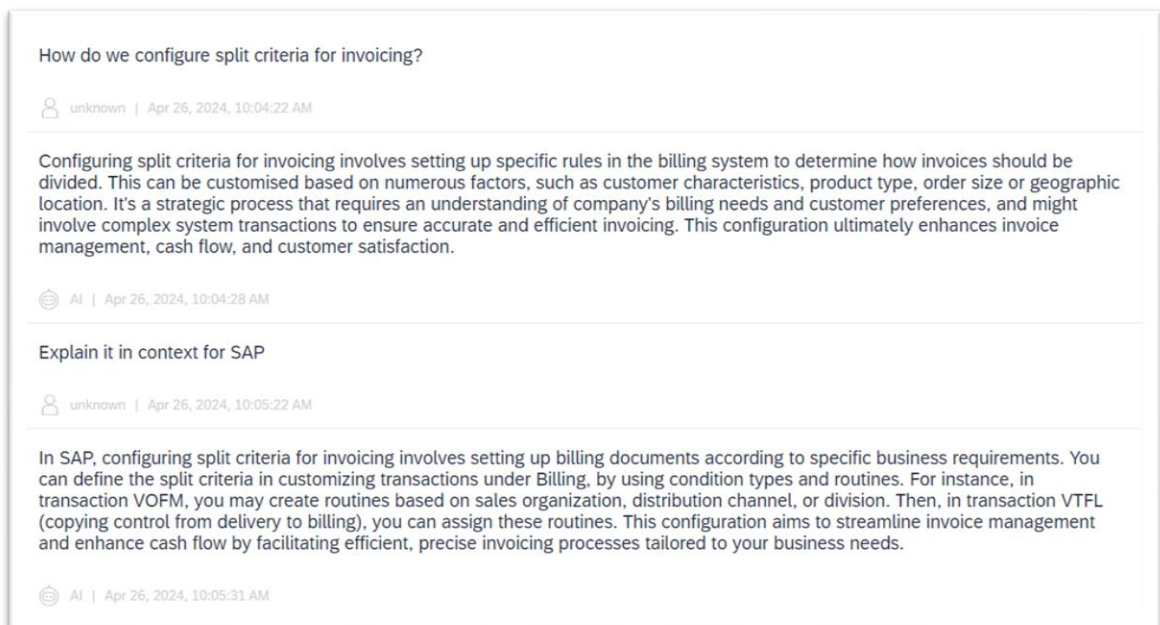


Figure 18 Providing context to a question

Addressing this issue could be straightforward – by incorporating the necessary context into the model's personality or by enabling the application to set the context either globally or allow users to specify it during each interaction. However, more complex challenges have emerged where simply adding context was insufficient. For example, in one instance, the model incorrectly continued to recommend the transaction code “XD02” for checking and altering debtor data when verifying a customer's tax classification. This advice would have been accurate in the context of SAP R/3, but in a more recent S/4HANA system, the transaction has been replaced by the new business partner concept.

This oversight might be excusable since the model was not informed that the system in use was S/4. However, even when directly questioned about the general applicability of the advice in an S/4 system, the model persisted with the outdated

transaction code, as shown in Figure 19. It was only after explicitly asking about the transaction “XD02” in the context of S/4HANA that the model acknowledged the mistake and provided the correct transaction code “BP”.

These examples underscore the importance of not only providing context but also ensuring that the model's responses are up-to-date with the latest system configurations and practices.

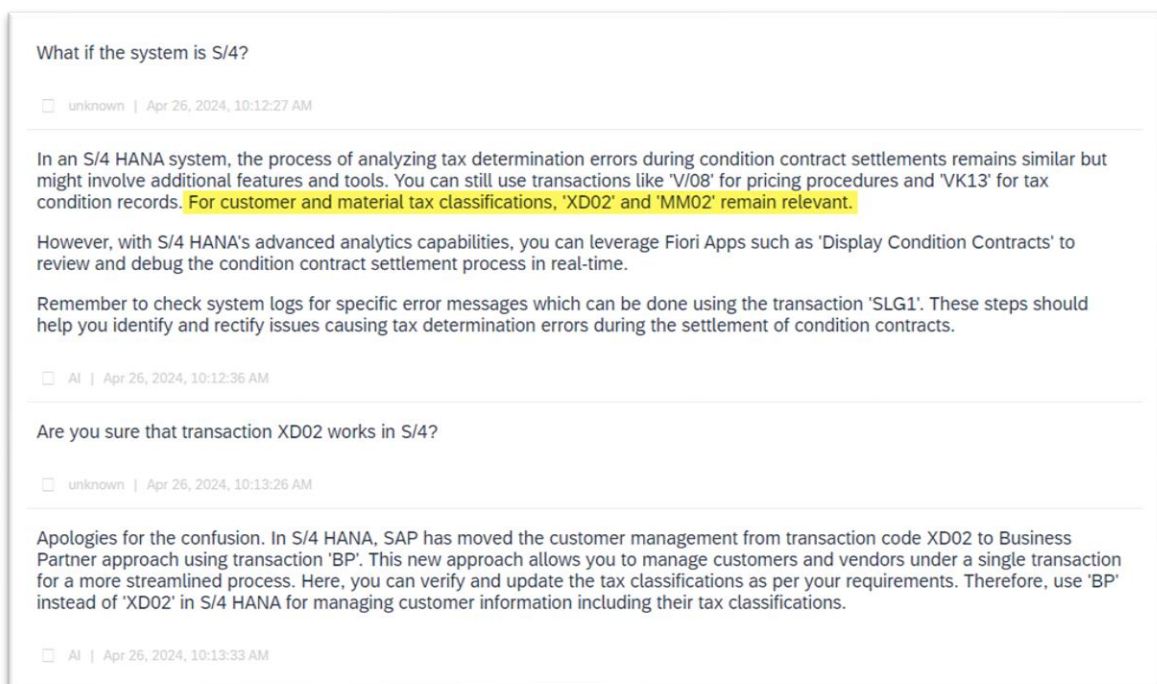


Figure 19 Model insisting on an incorrect answer

7.4.4.2 Access to Full Documentation

As previously discussed, direct access to a full document, without having to go through an LLM, offers some substantial advantages, such as the inclusion of images and a reduced likelihood of overlooking important details.

It may be helpful, therefore, to provide a link to the original documentation, based on which the answer was generated, alongside the response. This may include the customer-specific documentation, as well as internet documents – a functionality ChatGPT already provides, as demonstrated in Figure 20. With the current pace of innovation in the field, this may not be necessary in the future, however, as it may

also be possible at some point to include specific images embedded in the response.

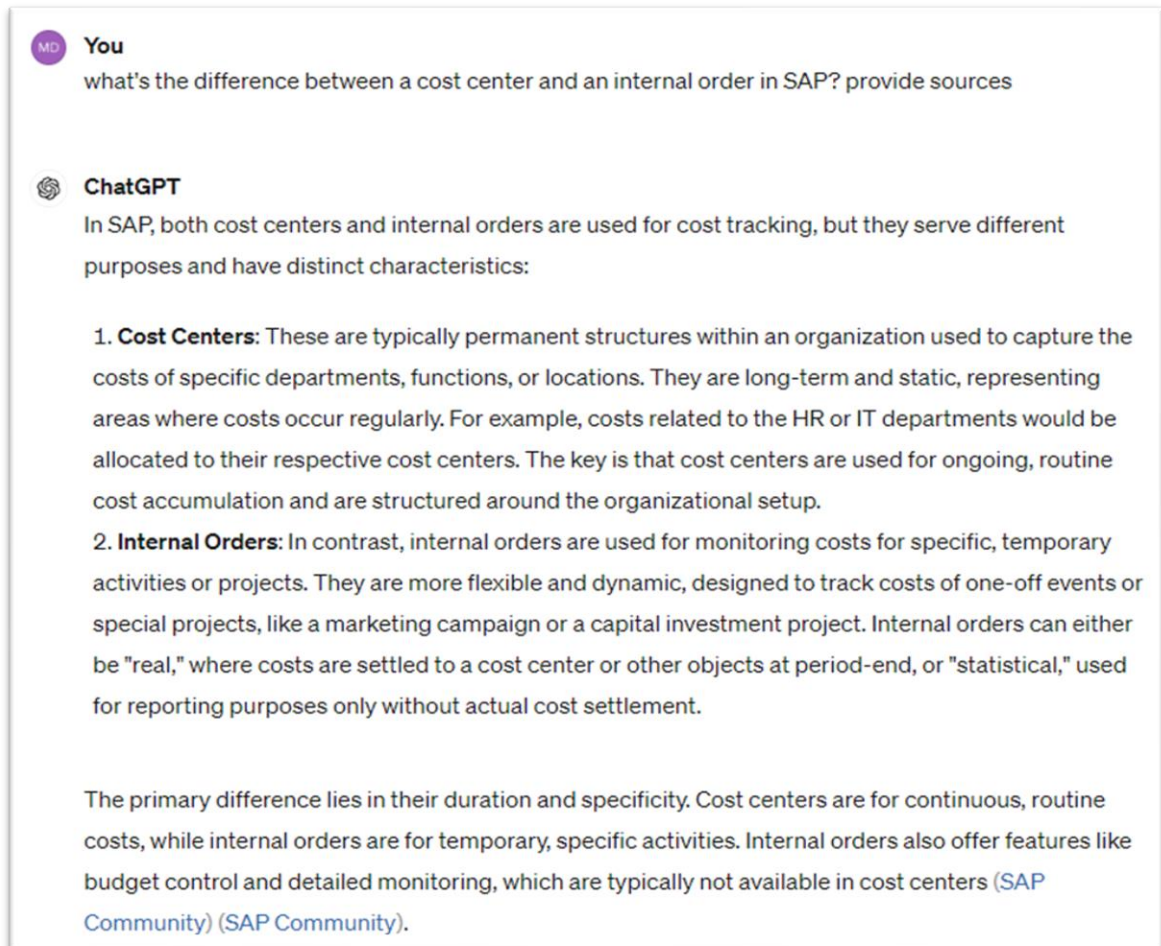


Figure 20 ChatGPT Browsing the Web and Providing Links to Sources

7.4.4.3 Continuous Improvement

A certain optimism about the future of such tools was expressed, anticipating continuous improvement and more precise and comprehensive answers in the future. This reflects a belief in the evolving capability of AI to assist in information retrieval and decision-making processes, compared to static documentation which is often times uploaded to a file share and then forgotten. (interview with Christian Haberfehlner, 2024, May 2, 96–99)

This could be achieved by providing users with the ability to give feedback on the model's responses. This feedback could then be used to fine-tune the model, alter

the “personalities” (as described in chapter 6.1), enhance the embedding algorithms or adapt the original documentation in order to update the data within or make it more easily processable by an LLM.

However, this may require resources only larger organizations may be able to muster, as it would most likely mean the establishment of a specific role for this task inside the organization. But if the aim is to maintain existing documentation and keeping it up to date with the help of an LLM, this might be the most important prerequisite.

7.4.4.4 Necessary User Skills

Leveraging a language model effectively may require users to develop and refine specific skills related to crafting prompts and asking questions. As discussed in the comparison with static documentation (refer to chapter 7.4.2) users may initially face challenges due to unfamiliarity with the necessary vocabulary or lack of a clear understanding of the topic at hand. This is a significant shift from typical search engine use, where the focus is often on very specific, keyword-driven queries, with results that may often be very comprehensive, unlike the initially general answers an LLM may provide.

For instance, when using an LLM, a user might start with a vague understanding of what information is needed. Unlike traditional search methods that operate well with direct and specific queries (“transaction for xyz”), LLMs can process and benefit from more open-ended and exploratory questions. To get the most out of an LLM, users should learn to first ask broad, foundational questions (“explain it to me first in general, give me the big picture”) to establish a context (interview with Mag. Oliver Szikonya, 2024, April 26, 71–76). This approach allows the LLM to provide a comprehensive backdrop, making subsequent, more specific queries more effective and easier to navigate.

However, as noted in the interviews, sometimes users might not even know the right questions to ask. This is particularly true for users who are new to a subject or are less experienced in formulating queries for AI-driven interactions. In these cases, there may be a need to ask follow-up questions or adjust queries based on the responses received, which further underscores the importance of developing a good questioning strategy. The ability to iteratively refine questions based on

previous answers is a critical skill that users need to develop to harness the full potential of LLMs. Thus, training or introductory guides on effective prompt-writing could greatly enhance user experience and outcomes when interacting with language models.

It will also be important for users to know when to trust the model's answers, which should depend on one's own know-how and ability to judge the correctness of the answers. (interview with Mag. Marius Treske, 2024, April 29, 29–35)

8. Conclusion

This thesis explored the potential of Large Language Models (LLMs) like GPT-4 in addressing the perpetual challenges associated with SAP implementations, specifically in managing legacy customizations and gaps in documentation. Through a comprehensive investigation that combines theoretical insights with practical applications, this research underscores the transformative potential of LLMs in enhancing the accessibility and utility of intricate ERP systems.

The introduction of a GPT-4 based chatbot prototype, aimed at answering the question of whether LLMs are able to effectively facilitate access to and maintenance of customer-specific information, represents a significant innovation in the field of SAP consultancy and usage.

The feedback from experts interviewed as part of this study largely underscores the adaptability and potential of the language model, particularly in its handling of complex, SAP-related content. Their insights reveal a nuanced picture of the model's performance, reflecting its robust knowledge base and its ability to process highly specific SAP-related documentation which often included material that was contradictory to the model's initial training data or that was not specifically formatted for the purpose of being parsed by a machine or in this case a large language model, demonstrating its ability to indeed bridge significant gaps in current documentation practices, providing intuitive, context-aware assistance that enhances both user understanding and operational efficiency.

However, the deployment of such advanced technology is not without its challenges. The thesis highlights the need for accurate context-setting in queries both by the chatbot application and the users themselves, and the continuous updating of model and application to handle the dynamic and complex nature of SAP systems. Moreover, the effectiveness of the LLM is contingent upon the user's ability to formulate precise questions and engage with the model's responses critically, possibly necessitating in-depth user trainings.

Additionally, a solution of this kind will yield the best results when it is not seen in the context of a ‘fire-and-forget’ paradigm, but rather as part of a continuous improvement process. By incorporating feedback mechanisms for users, the model can be fine-tuned and documentation adapted to make it more accessible for an LLM.

Moreover, this process of continuous improvement is a prerequisite to second aspect of the research question, namely the maintenance of existing documentation. This may only be feasible for larger organizations, however, as it may require considerable resources and ongoing effort – especially in the beginning of the rollout but also down the line.

In conclusion, while the LLM-based solutions show great promise in revolutionizing SAP documentation and user support, they require careful implementation and ongoing refinement to fully realize their potential and may not yet be able to fully replace static documentation. This study not only contributes valuable insights into the integration of LLMs within SAP environments but also sets a foundation for future research and development in this area. As such, this work is a pivotal step towards more streamlined, accessible, and user-friendly management of SAP systems, ultimately enhancing the effectiveness of ERP implementations across diverse organizational landscapes.

9. References

- Adnette Kamugisha. 2021. “SAP Activate Methodology for RISE with SAP S/4HANA Cloud, Private Edition | SAP Blogs.” Accessed December 20, 2023. <https://blogs.sap.com/2021/02/08/rise-with-sap-sap-activate-methodology-for-with-sap-s-4hana-cloud-private-edition/>.
- Amaratunga, Thimira. 2023. *Understanding Large Language Models: Learning Their Underlying Concepts and Technologies*. 1st ed. 2023. Berkeley, CA: Apress; Imprint Apress.
- Anderson, George W. 2009. *SAP Implementation Unleashed: A Business and Technical Roadmap to Deploying SAP*. Indianapolis, Ind., London: Sams; Pearson Education.
- Angelis, Luigi de, Francesco Baglivo, Guglielmo Arzilli, Gaetano Pierpaolo Privitera, Paolo Ferragina, Alberto Eugenio Tozzi, and Caterina Rizzo. 2023. “ChatGPT and the Rise of Large Language Models: The New AI-Driven Infodemic Threat in Public Health.” *Front. Public Health* 11:1166120. <https://doi.org/10.3389/fpubh.2023.1166120>.
- Azaria, Amos, and Tom Mitchell. 2023. “The Internal State of an Llm Knows When Its Lying.” *arXiv preprint arXiv:2304.13734*.
- Banks-Grasedyck, Denise, Eckhard Lippke, Hans Oelfin, Reinhold Schwaiger, and Volker Seemann. 2022. *Successfully Managing S/4HANA Projects: The Definitive Guide to the Next Digital Transformation*. Springer eBook Collection. Cham: Springer International Publishing; Imprint Springer.
- Bardhan, Devraj. 2021. *SAP S/4HANA: An Introduction*. With the assistance of A. Baumgartl, N.-S. Choi, M. Dudgeon, P. Górecki, A. Lahiri, B. Meijerink and A. Worsley-Tonks. 4th ed. Boston: Rheinwerk Publishing Inc. <https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=6884502>.
- Becker, Brett A., Paul Denny, James Finnie-Ansley, Andrew Luxton-Reilly, James Prather, and Eddie Antonio Santos. 2023. “Programming Is Hard - or at Least It Used to Be.” In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, edited by Maureen Doyle, 500–506. ACM Digital Library. New York, NY, United States: Association for Computing Machinery.

- Bengio, Yoshua, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. "A Neural Probabilistic Language Model." *J. Mach. Learn. Res.* 3 (null): 1137–55.
- Bortz, Jürgen, and Nicola Döring. 2006. "Qualitative Methoden." In *Forschungsmethoden Und Evaluation: Für Human- Und Sozialwissenschaftler*, 295–350. Berlin, Heidelberg: Springer Berlin Heidelberg.
- Boyd, Eric. 2023. "Introducing GPT-4 in Azure OpenAI Service." *Microsoft Azure Blog*, March 21, 2023. <https://azure.microsoft.com/en-us/blog/introducing-gpt4-in-azure-openai-service/>.
- Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan et al. 2020. "Language Models Are Few-Shot Learners." *Advances in Neural Information Processing Systems* 33: 1877–1901.
- Candito, Nick. 2016. "How Inefficient Processes Are Hurting Your Company." *Entrepreneur*, December 8, 2016. <https://www.entrepreneur.com/growing-a-business/how-inefficient-processes-are-hurting-your-company/286084>.
- Christian Haberehner. 2024. Interview by Martin Drzola. May 2, 2024. Vienna.
- Danks, David, and Alex John London. 2017. "Regulating Autonomous Systems: Beyond Standards." *IEEE Intelligent Systems* 32 (1): 88–91.
- Dave, Ravi, Bidyut Sarkar, and Gaurav Singh. 2023. "Revolutionizing Business Processes with SAP Technology: A Buyer's Perspective." *International Journal of Computer Trends and Technology* 71 (4): 1–7.
- Densborn, Frank, and Frank Finkbohner. 2021. *Migrating to SAP S/4HANA*. 2nd edition. Boston: Rheinwerk Publishing.
- Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. *BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding*.
- Duan, Haonan, Adam Dziedzic, Mohammad Yaghini, Nicolas Papernot, and Franziska Boenisch. 2023. "On the Privacy Risk of in-Context Learning." Accessed January 01, 2024. <https://trustnlpworkshop.github.io/papers/13.pdf>.

- Egli, Adrian. 2023. "ChatGPT, GPT-4, and Other Large Language Models: The Next Revolution for Clinical Microbiology?" *Clin Infect Dis* 77 (9): 1322–28. <https://doi.org/10.1093/cid/ciad407>.
- Elias, Jennifer. 2023. "Google's Newest A.I. Model Uses Nearly Five Times More Text Data for Training Than Its Predecessor." Accessed January 03, 2024. <https://www.cnn.com/2023/05/16/googles-palm-2-uses-nearly-five-times-more-text-data-than-predecessor.html>.
- Fesko, Sean. 2022. "What Is RISE with SAP?" Accessed December 20, 2023. <https://blog.sap-press.com/what-is-rise-with-sap>.
- Franceschelli, Giorgio, and Mirco Musolesi. 2023. "On the Creativity of Large Language Models." <http://arxiv.org/pdf/2304.00008.pdf>.
- Gat, Yair, Nitay Calderon, Amir Feder, Alexander Chapanin, Amit Sharma, and Roi Reichart. 2023. *Faithful Explanations of Black-Box NLP Models Using LLM-Generated Counterfactuals*.
- Guidotti, Riccardo, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2019. "A Survey of Methods for Explaining Black Box Models." *ACM Comput. Surv.* 51 (5): 1–42. <https://doi.org/10.1145/3236009>.
- Guo, Jiayue, Wenqian Zhang, and Tiansheng Xia. 2023. "Impact of Shopping Website Design on Customer Satisfaction and Loyalty: The Mediating Role of Usability and the Moderating Role of Trust." *Sustainability* 15 (8): 6347. <https://doi.org/10.3390/su15086347>.
- Hochreiter, S., and J. Schmidhuber. 1997. "Long Short-Term Memory." *Neural Computation* 9 (8): 1735–80. <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Holub, Ilja, and Tomas Bruckner. 2016. "Measuring Complexity of SAP Systems." *CSIMQ* 0 (8): 60–67. <https://doi.org/10.7250/csimq.2016-8.05>.
- Huang, Hanyao, Ou Zheng, Dongdong Wang, Jiayi Yin, Zijin Wang, Shengxuan Ding, Heng Yin et al. 2023. "ChatGPT for Shaping the Future of Dentistry: The Potential of Multi-Modal Large Language Model." *Int J Oral Sci* 15 (1): 29. <https://doi.org/10.1038/s41368-023-00239-y>.

- Hummel, Patrik, Matthias Braun, and Peter Dabrock. 2021. "Own Data? Ethical Reflections on Data Ownership." *Philos. Technol.* 34 (3): 545–72.
<https://doi.org/10.1007/s13347-020-00404-9>.
- Ippolito, Daphne, Florian Tramèr, Milad Nasr, Chiyuan Zhang, Matthew Jagielski, Katherine Lee, Christopher A. Choquette-Choo, and Nicholas Carlini. 2023. *Preventing Verbatim Memorization in Language Models Gives a False Sense of Privacy*.
- Irsoy, Ozan, and Claire Cardie. 2014. "Deep Recursive Neural Networks for Compositionality in Language." In *Advances in Neural Information Processing Systems*. Vol. 27, edited by Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger: Curran Associates, Inc.
https://proceedings.neurips.cc/paper_files/paper/2014/file/2cfd4560539f887a5e420412b370b361-Paper.pdf.
- Islam, Md. Aminul. "SAP FICO in Relation with Order Delivery Process in Enterprise Resource Planning : A Case Study on SAP S/4HANA System Environment as a Practical Platform."
<https://www.theseus.fi/handle/10024/227967>.
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama et al. 2022. "Emergent Abilities of Large Language Models." *Transactions on Machine Learning Research*.
<https://openreview.net/forum?id=yzkSU5zdwD>.
- Ji, Ziwei, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. "Survey of Hallucination in Natural Language Generation." *ACM Comput. Surv.* 55 (12): 1–38.
<https://doi.org/10.1145/3571730>.
- Jonas Geiping, and Tom Goldstein. 2023. "Cramming: Training a Language Model on a Single GPU in One Day." *International Conference on Machine Learning*, 11117–43. <https://proceedings.mlr.press/v202/geiping23a.html>.
- Jurafsky, Dan, and James H. Martin. 2023. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Accessed December 27, 2023.
<https://web.stanford.edu/~jurafsky/slp3/>.

- Kaplan, Jared, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. "Scaling Laws for Neural Language Models." <http://arxiv.org/pdf/2001.08361v1>.
- Kerim Goztepe, Muammer Karaman, and Hayrettin Catalkaya. 2015. *Fuzzy Rule-Based Approach for Enterprise Resource Planning (ERP) Software Evaluation* 11. https://www.researchgate.net/publication/281748662_Fuzzy_Rule-Based_Approach_for_Enterprise_Resource_Planning_ERP_Software_Evaluation.
- Kerkmann, Christof. 2023. "SAP: Warum Viele Kunden Mit Der Cloud-Strategie Zögern." *Handelsblatt*, September 25, 2023. <https://www.handelsblatt.com/technik/it-internet/sap-warum-viele-kunden-mit-der-cloud-strategie-zoegern/29399334.html>.
- Kim, Tae-Young, and Sung-Bae Cho. 2019. "Predicting Residential Energy Consumption Using CNN-LSTM Neural Networks." *Energy* 182:72–81. <https://doi.org/10.1016/j.energy.2019.05.230>.
- Kung, Po-Nien, and Nanyun Peng. 2023. *Do Models Really Learn to Follow Instructions? An Empirical Study of Instruction Tuning*.
- Lange, Jörg, Frank-Peter Bauer, Christoph Persich, Tim Dalm, Gunther Sanchez, and Tobias Adler. 2020. *Warehouse Management Mit SAP® EWM*. 4., aktualisierte und erweiterte Auflage. Rheinwerk Publishing. Bonn: Rheinwerk Verlag. <https://permalink.obvsg.at/AC15539906>.
- Lewis, Mike, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. *BART: Denoising Sequence-to-Sequence Pre-Training for Natural Language Generation, Translation, and Comprehension*.
- Li, Zewen, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. 2022. "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects." *IEEE Trans. Neural Netw. Learning Syst.* 33 (12): 6999–7019. <https://doi.org/10.1109/tnnls.2021.3084827>.
- Lin, Tianyang, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2022. "A Survey of Transformers." *AI Open* 3:111–32. <https://doi.org/10.1016/j.aiopen.2022.10.001>.

- Lyu, Qing, Marianna Apidianaki, and Chris Callison-Burch. 2023. *Towards Faithful Model Explanation in NLP: A Survey*.
- Mag. Marius Treske. 2024. Interview by Martin Drzola. April 29, 2024. Vienna.
- Mag. Oliver Szikonya. 2024. Interview by Martin Drzola. April 26, 2024. Vienna.
- Managing Business with SAP: Planning, Implementation and Evaluation*. 2005. EBSCOhost eBook Collection. Hershey, PA: Idea Group Pub.
<http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=108379>.
- Mao, Huanru Henry. 2020. *A Survey on Self-Supervised Pre-Training for Sequential Transfer Learning in Neural Networks*.
- Martyn, J., and B. C. Vickery. 1970. "The Complexity of the Modelling of Information Systems." *Journal of Documentation* 26 (3): 204–20.
<https://doi.org/10.1108/eb026494>.
- Mayring, Philipp. 2000. "Qualitative Content Analysis." *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, Vol 1, No 2 (2000): Qualitative Methods in Various Disciplines I: Psychology. *FQS* 1 (2).
<https://doi.org/10.17169/fqs-1.2.1089>.
- Mayring, Philipp. 2015. *Qualitative Inhaltsanalyse: Grundlagen und Techniken*. 12., überarbeitete Auflage. Beltz Pädagogik. Weinheim, Basel: Beltz.
http://content-select.com/index.php?id=bib_view&ean=9783407293930.
- McIntosh, Timothy R., Tong Liu, Teo Susnjak, Paul Watters, Alex Ng, and Malka N. Halgamuge. 2024. "A Culturally Sensitive Test to Evaluate Nuanced GPT Hallucination." *IEEE Trans. Artif. Intell.*, 1–13.
<https://doi.org/10.1109/TAI.2023.3332837>.
- Meskó, Bertalan, and Eric J. Topol. 2023. "The Imperative for Regulatory Oversight of Large Language Models (Or Generative AI) In Healthcare." *npj Digit. Med.* 6 (1): 120. <https://doi.org/10.1038/s41746-023-00873-0>.
- Meta*. 2023. "Meta and Microsoft Introduce the Next Generation of Llama." July 18, 2023. <https://about.fb.com/news/2023/07/llama-2/>.

- Mike Zaschka. 2023a. "A ChatGPT-Like Chat App Built with Modern SAP Technologies (CAP, SAPUI5)." Accessed April 27, 2024.
<https://community.sap.com/t5/technology-blogs-by-members/a-chatgpt-like-chat-app-built-with-modern-sap-technologies-cap-sapui5-part/ba-p/13549057>.
- Mike Zaschka. 2023b. *Cap-Ui5-Gpt-Chat*. GitHub. Accessed April 27, 2024.
<https://github.com/p36-io/cap-ui5-gpt-chat>.
- Mirchandani, Vinnie. 2014. *SAP Nation: A Runaway Software Economy*. Tampa, FL: Deal Architect, Inc.
- Mireshghallah, Fatemehsadat, Archit Uniyal, Tianhao Wang, David Evans, and Taylor Berg-Kirkpatrick. 2022. *Memorization in NLP Fine-Tuning Methods*.
- Mutegeki, Ronald, and Dong Seog Han. 2020. "A CNN-LSTM Approach to Human Activity Recognition." In *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*: IEEE.
- Naveed, Humza, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. *A Comprehensive Overview of Large Language Models*.
- Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. "Large Language Models Struggle to Learn Long-Tail Knowledge." *International Conference on Machine Learning*, 15696–707.
<https://proceedings.mlr.press/v202/kandpal23a.html>.
- Oliver Graeff. 2021. "What Is SAPUI5?" Accessed May 01, 2024.
<https://community.sap.com/t5/technology-blogs-by-sap/what-is-sapui5/ba-p/13504830>.
- Omiye, Jesutofunmi A., Haiwen Gui, Shawheen J. Rezaei, James Zou, and Roxana Daneshjou. 2023. *Large Language Models in Medicine: The Potentials and Pitfalls*.
https://www.researchgate.net/publication/373642018_Large_language_models_in_medicine_the_potentials_and_pitfalls.
- OpenAI. 2023a. "Introducing ChatGPT Enterprise: Get Enterprise-Grade Security & Privacy and the Most Powerful Version of ChatGPT yet." Accessed January 04, 2024. <https://openai.com/blog/introducing-chatgpt-enterprise>.

- OpenAI. 2023b. “New Ways to Manage Your Data in ChatGPT.” Accessed January 04, 2024. <https://openai.com/blog/new-ways-to-manage-your-data-in-chatgpt>.
- OpenAI. 2024a. “GPT-4 Turbo and GPT-4: OpenAI API.” Accessed May 03, 2024. <https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4>.
- OpenAI. 2024b. “What Are Tokens and How to Count Them?” Accessed April 27, 2024. <https://help.openai.com/en/articles/4936856-what-are-tokens-and-how-to-count-them>.
- Peter M. Färbing. 2023. “SAP-Cloud-Strategie: Warum Sie Für Viele SAP-Anwender so Problematisch Ist.” *heise online*, November 8, 2023. <https://www.heise.de/hintergrund/SAP-Cloud-Strategie-Warum-sie-fuer-viele-SAP-Anwender-so-problematisch-ist-9356272.html>.
- Prather, James, Brent N. Reeves, Paul Denny, Brett A. Becker, Juho Leinonen, Andrew Luxton-Reilly, Garrett Powell, James Finnie-Ansley, and Eddie Antonio Santos. 2024. ““It’s Weird That It Knows What I Want”: Usability and Interactions with Copilot for Novice Programmers.” *ACM Trans. Comput.-Hum. Interact.* 31 (1): 1–31. <https://doi.org/10.1145/3617367>.
- Preuss, Peter. 2017. *In-Memory-Datenbank SAP HANA*. Wiesbaden: Springer Fachmedien Wiesbaden.
- Ramlochan, Sunil. 2023. “How Does Llama-2 Compare to GPT-4/3.5 and Other AI Language Models.” *Prompt Engineering*, September 2, 2023. <https://promptengineering.org/how-does-llama-2-compare-to-gpt-and-other-ai-language-models/>.
- Ray, Partha Pratim. 2023. “ChatGPT: A Comprehensive Review on Background, Applications, Key Challenges, Bias, Ethics, Limitations and Future Scope.” *2667-3452* 3:121–54. <https://doi.org/10.1016/j.iotcps.2023.04.003>.
- Ruder, Sebastian. 2018. *A Review of the Neural History of Natural Language Processing*. Accessed December 27, 2023. <https://www.ruder.io/a-review-of-the-recent-history-of-nlp/>.
- Sadikoğlu, Emre, Murat GÖK, Maad M. MIJWIL, and İrfan Kösesoy. 2023. “The Evolution and Impact of Large Language Model Chatbots in Social Media: A

- Comprehensive Review of Past, Present, and Future Applications.” *Veri Bilim Derg* 6 (2): 67–76. <https://dergipark.org.tr/en/pub/veri/issue/81532/1400734>.
- SAP SE. 2022. “SAP Fiori Launchpad: SAP Fiori for Web Design Guidelines.” Accessed May 01, 2024. <https://experience.sap.com/fiori-design-web/launchpad/>.
- SAP SE. 2023. “ABAP - Keyword Documentation.” Accessed December 18, 2023. https://help.sap.com/doc/abapdocu_752_index_htm/7.52/en-us/abenabap.htm.
- Sayer, Peter. 2024. “SAP Doubles down on Cloud-First Innovation with Executive Reshuffle.” *CIO*, January 10, 2024. <https://www.cio.com/article/1289836/sap-doubles-down-on-cloud-first-innovation-with-executive-reshuffle.html>.
- Schmidt, Vasco Alexander. 2005. “Technisches Schreiben bei SAP.” In *Schreiben am Arbeitsplatz*, edited by Katrin Lehnen, Kirsten Schindler, and Eva-Maria Jakobs, 73–92. SpringerLink Bücher. Wiesbaden: VS Verlag für Sozialwissenschaften. https://link.springer.com/chapter/10.1007/978-3-322-80777-9_5.
- Schreiner, Maximilian. 2023. “GPT-4 Architecture, Datasets, Costs and More Leaked.” *THE-DECODER.com*, July 11, 2023. <https://the-decoder.com/gpt-4-architecture-datasets-costs-and-more-leaked/>.
- Simonyan, Karen, and Andrew Zisserman. 2014. “Very Deep Convolutional Networks for Large-Scale Image Recognition.” <http://arxiv.org/pdf/1409.1556v6>.
- Singh, Vinay. 2017. *Manage Your SAP Projects with SAP Activate: Implementing SAP S/4HANA*. 1st ed. Birmingham, United Kingdom: Packt Publishing Limited. <https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=5090761>.
- Strübing, Jörg, ed. 2018. *Qualitative Sozialforschung: Eine komprimierte Einführung*. 2., überarbeitete und erweiterte Auflage. Soziologie kompakt. Berlin, Boston: De Gruyter Oldenbourg.
- Sven Denecken. 2020. “The Power of Best-in-Class and Next Practices Business Content for a Modern ERP.” Accessed December 19, 2023. <https://blogs.sap.com/2020/05/12/the-power-of-best-in-class-and-next-practices-business-content-for-a-modern-erp/>.

- Ted Sanders, and Mike Heaton. 2022. "Question Answering Using Embeddings-Based Search." Accessed April 30, 2024.
https://cookbook.openai.com/examples/question_answering_using_embeddings.
- Thoppilan, Romal, Daniel de Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin et al. 2022. "LaMDA: Language Models for Dialog Applications." <http://arxiv.org/pdf/2201.08239.pdf>.
- Toyama, Yoshitaka, Ayaka Harigai, Mirei Abe, Mitsutoshi Nagano, Masahiro Kawabata, Yasuhiro Seki, and Kei Takase. 2023. "Performance Evaluation of ChatGPT, GPT-4, and Bard on the Official Board Examination of the Japan Radiology Society." *Jpn J Radiol*, 1–7. <https://doi.org/10.1007/s11604-023-01491-2>.
- Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." *Advances in Neural Information Processing Systems* 30.
<https://proceedings.neurips.cc/paper/7181-attention-is-all>.
- Wang, Jin, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2016. "Dimensional Sentiment Analysis Using a Regional CNN-LSTM Model." In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Stroudsburg, PA, USA: Association for Computational Linguistics.
- Wang, Junjie, Yuchao Huang, Chunyang Chen, Zhe Liu, Song Wang, and Qing Wang. 2023. "Software Testing with Large Language Model: Survey, Landscape, and Vision." <http://arxiv.org/pdf/2307.07221.pdf>.
- Wegelin, Michael, and Michael Englbrecht. 2018. *SAP-Schnittstellenprogrammierung*. 4. Auflage, erweiterte Ausgabe. Bonn: Rheinwerk Verlag; SAP PRESS.
- Xu, Rongwu, Zehan Qi, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024. "Knowledge Conflicts for LLMs: A Survey." <http://arxiv.org/pdf/2403.08319>.

Zhang, Shengyu, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li et al. 2023. *Instruction Tuning for Large Language Models: A Survey*.

Zhao, Wayne Xin, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min et al. 2023. *A Survey of Large Language Models*.

Zhou, Kun, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. 2023. *Don't Make Your LLM an Evaluation Benchmark Cheater*.

10. Index of Abbreviations

ERP	Enterprise Resource Planning
LLM	Large Language Model
BTP	Business Technology Platform
CAP	Cloud Application Programming Model
CNN	Convolutional neural network
RNN	Recurrent neural network
PLM	Pre-trained language model
LSTM	Long Short-Term Memory network

11. Appendix

11.1 Interview Guide



Interview_Guide.pdf

11.2 Electronic Bank Statement Customer Documentation



Documentation_EB
S.pdf

11.3 Interview Mag. Oliver Szikonya



Oliver_20240426.pdf

11.4 Interview Mag. Marius Treske



Marius_20240429.pdf

11.5 Interview Christian Haberfehlner



Christian_20240502.
pdf