Conversational AI. Dialogsysteme, Chatbots, Assistenten

Veranstalter: Christoph Ringlstetter

Sitzung 6: Memory

Was machen wir denn heute.

- Orga. Referate, Zulassung, Termine.
- News of the Week
- Referat Chatbots als persönliche Begleiter
- Besprechung Paper Hello Again
- Besprechung Paper Agents with context and time sensitive memory
- Nochmal Hinweis zum Langchain Memory in den Tutorials

Hello Again…HaoLi et al – Überblick Paper

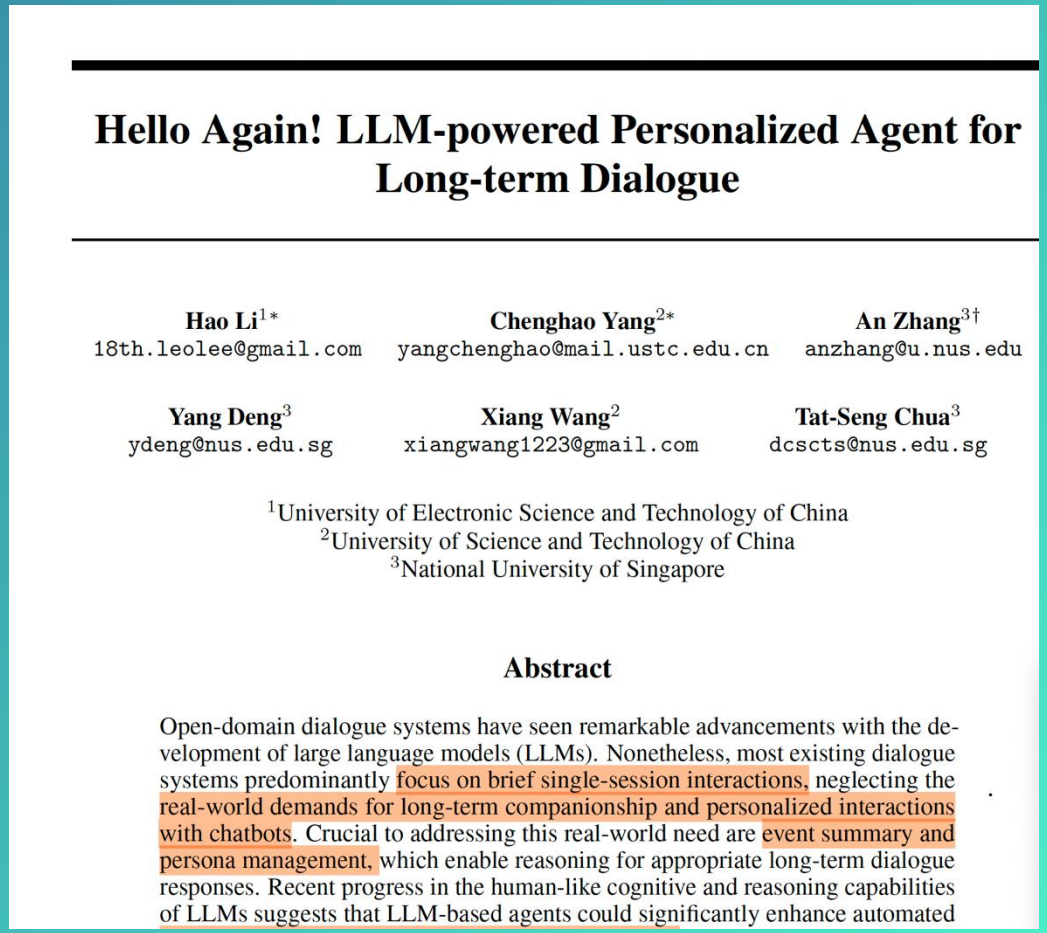
Viele Chatbots nur für kurze, single session Interaktionen brauchbar.

Nachfrage: longterm companionship.

Personalisierte Interaktionen → brauchen Event Summary und Persona Management

Verbesserungen: Wahrnehmung, Entscheidungen, Problemlösungsfähigkeit

Module des LongTermDialogAgent: Event Wahrnehmung, Persona Extraktion, Antwort Generierung.



Hello Again! LLM-powered Personalized Agent for Long-term Dialogue

Hao Li[1*] 18th.leolee@gmail.com

Chenghao Yang[2*] yangchenghao@mail.ustc.edu.cn

An Zhang[3†] anzhang@u.nus.edu

Yang Deng[3] ydeng@nus.edu.sg

Xiang Wang[2] xiangwang1223@gmail.com

Tat-Seng Chua[3] dcscts@nus.edu.sg

[1]University of Electronic Science and Technology of China
[2]University of Science and Technology of China
[3]National University of Singapore

Abstract

Open-domain dialogue systems have seen remarkable advancements with the development of large language models (LLMs). Nonetheless, most existing dialogue systems predominantly focus on brief single-session interactions, neglecting the real-world demands for long-term companionship and personalized interactions with chatbots. Crucial to addressing this real-world need are event summary and persona management, which enable reasoning for appropriate long-term dialogue responses. Recent progress in the human-like cognitive and reasoning capabilities of LLMs suggests that LLM-based agents could significantly enhance automated

Hello Again…HaoLi et al – Überblick Paper

These des Papers: Open Domain Dialogsysteme zielen auf eine langfristige, personalisierte Interaktion ab.

=> Chatbot muss extensive Dialoghistorie erinnern, reflektieren in das „Gespräch“ einordnen und ausserdem permanent die Charakteristika des Users und seine eigenen updaten: Persona update.

Hao Li. S.2

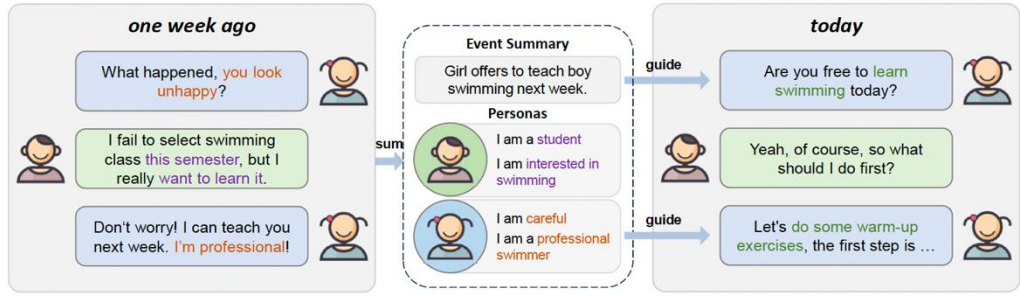


Figure 1: The illustration of how event memory and personas guide long-term dialogue. The event summary and personas are extracted from a conversation that occurred one week ago. In today's interaction, the event memory prompts the girl to inquire about the swimming lesson they scheduled last week. The personas, indicating that she is careful and professional in swimming, guide her to offer detailed and professional advice.

Hello Again…HaoLi et al – Überblick Paper

Herausforderung: Gleichzeitig ein longterm even memory und die Persona konsistent halten.

Strategie SOTA: Trennen von Persona und Event Memory.

Memory:

→ memory bank um events zu speichern.

→ RAG Ansatz um relevante Information zuzugreifen und Antworten zu generieren

Persona:

→ unidirektionales User Modelling

→ bidirektionales Agent—User Modelling Lit [15,16,3] noch nachlesen

Hello Again…HaoLi et al – Überblick Paper

=> Nachteil ist die Abhängigkeit von Modellarchitekturen: keine Adaption an neue Modelle.
=> Die Dialogmodelle haben mangelhafte Zero-Shot Generalisierung → Probleme mit den Realworld Domains.

Ziel: optimiertes longterm Dialogframework: model agnonstic, deployable auf Realworld Domänen. Autonom fähig Daten aus den Event Memories und den Personas zu integrieren.

Hello Again…HaoLi et al – Überblick Paper

Lösungspfad: LLM hat exzellente menschenähnliche kognitive und reasoning Fähigkeiten => LLM ist Kern der agentenbasierten Simultationen
Automatisieren: Perception, Decision Making, Problem Solving

LDAgent: Model Agnostic Framework: Event Memory, Wahrnehmungsmodul, Persona Extractor, Antwortgenerierungsmodul.

Hello Again…HaoLi et al – Überblick Paper

Elemente:

Event Memory: Kohärenz über die Sessions hinweg. Shortterm, longterm.

Shortterm: kontextuelle Information zur ongoing Konversation

Longterm: Vektordatenbank mit Highlevel Event Summaries.

Persona: personalisierte Interaktion ermöglichen. Justierbarer Mechanismus zum user-agent Modelling. Longterm Persona Bank als Tool.

Die Persona-Info wird dann zusammen mit den Memories in die Response Generation eingebracht.

Experimente auf: MSC, Conversation Chronicles: Korpora für multisession longterm Dialoge die angepasst werden

Hello Again…HaoLi et al – Methoden

Task Definition: Ziel der long-term multi-session Dialogaufgabe: passende Antwort r unter Berücksichtigung von Session Kontext C mit ausgewählten Informationen aus historischen Sessions H.

C: {u_1,u_2…. ud_{c-1}, u_{dc}} mit dc Turns der gegenwärtigen session.
In H: N historische Sessions: H: {h_1,h_2….h_{di} } Zahl der Äußerungen in Historischen Sessions. → ?? Was heisst das?? Sind die H flach gespeichert.

Hello Again…HaoLi et al – Methoden

Event Perception: Wahrnehmung über shortterm, longterm memory

Longterm: Memory Storage für das kodieren und extrahieren von Events aus früheren Sessions.

occurrence Time t.

Event Summary o.

Kurzes Summary: o in Speicherbank M_L { $\Theta(t_j,o_j) | j \in \{1,2,..l\}$ }

$\Theta(.)$: Text encoder zB MiniLM Lit [28]

Retrieval per Embedding Mechanismus

Hello Again…HaoLi et al – Methoden

Event Summary: Vorgehen [20,21,29] in der Agent community wie folgt:
LLM zero shot. Herausschneiden und Summarize Events aus einer Kommunikation
=> hierzu instruction tuning [30]
auf das Summary Modul anwenden.
Speziell: DialogSum Datenset angepasst. Ein Datensatz besteht aus:
(1) task background
(2) related conversations => into LLM as Prompt => Dialog Sum Antwort
(3) Summarization requests als Objective

wird dann für Finetuning benutzt

Hello Again…HaoLi et al – Methoden

Memory Retrieval: Verbessern der Retrieval Qualität

Semantic Relevance, Topic overlap, Time Decay.

Retrieval Schritt: wie die relevanten Memorydaten bekommen. Methoden bislang [20,21] benutzen Event Summaries als Keys und Context als Queries.

S_{sem}: Score

=> signifikante Fehler.

Idee im Paper: Erhöhen der Retrieval Quality.

Extraktion von Nomen aus den korrespondierenden Konversationen und den Summaries: Topiclexikon.

Dann: Overlap Scores:

Hello Again…HaoLi et al – Methoden

Overlap Scores

S_{topic} = ½ |Vq ∩ Vk|/|Vq| + |Vq ∩Vk|/|Vk| (im Paper fehlen im Nenner die Betragsstriche prüfen??)

Formel im Paper verstehe ich nicht. V sind die topic Nomen Sets in Query und den Retrievalergebnissen.

Time decay: Coefficent per e-Funktion hoch –t

Dann noch eine Threshold für den Score um irrelevante Memories auszuschliesen.

Insgesamt: sehr komplexer Formelaufbau: Overfitting?

Hello Again…HaoLi et al – Methoden

Short Term Memory:

Aktiver, dynamischer Dialog Cache: $M_s = \{ (t_i,u_i) | i=\{1…r_c\}\}$

Es kommt eine Äußerung u‘ rein. Das Modell evaluiert zuerst die Zeitdifferenz zwischen current time t‘ und der zuletzt aufgenommenen Zeit t_{rc} im Chache. Falls das Intervall eine Treshold β überschreitet → Trigger an das Longtermmemory um die chached Einträge an die Summary Funktion zu senden und einen neuen Eintrag in M_L zu generieren. Clear M_sChache.

(t‘,u‘) → chache M_s

Hello Again…HaoLi et al – Methoden

Dynamische Persona Extraktion: Persona Modul: soll die long-term Persona Konsistenz sichern. Für beide Teilnehmer in einem Dialogsystem Agent + Human [3].

Bidirektionaler User-Agent Ansatz neu im Paper.

Anpassbarer (tunable) Persona Extraktor der auf einen Persona Datenbank P_u P_a zugreift.

Open Domain Ausserungsbasiertes Persona Extraction Set vom Paper bereitgestellt: abgeleitet aus MSC [1]

Anreicherung/Tuning: mit LORA basiertem Instruction Tuning

Dann möglich: Extraktion von Persönlichkeitsmerkmalen aus dem Chatverlauf

Hello Again…HaoLi et al – Methoden

Die Persönlichkeitsmerkmale werden nach und nach in der Persönlichkeitsdatenbank zu dem Charakter abgespeichert.
Auch möglich: tuningfree: über Zeroshot LLM personas basierend auf prompts extrahieren. Besser: Chain of Thought Prompting.

Hello Again…HaoLi et al – Methoden

Response Generation: u‘ User Äusserung kommt rein:

1) retrieve relevant long term memories

2) short term context laden

3) bringe Personas Pa, Pu online

1-3 werden in den Response generator zusammen mit der Äußerung eingebracht.

Generator Tuning: longterm, multi-session, Dialogdatenset. Dynamische Retrieval Memories, Kontext Personas aus MSC und CC Datensets als Groundtruth.

Hello Again…HaoLi et al – Experimente

5 Sessions aus je MSC, CC mit je 50 turns. Das ist nicht allzuviel.

Experimente zu Modellunabhängigkeit, Abschaltung einzelner Module, Persona Extraktor und Crossdomain.

Metriken:

BLEU-N (BL-N) [33], ROUGE-L (R-L) [34], and METEOR (MET) [35]

Antwortgenerierung

(ACC): Klassifikation Persona Extractor

Human Evaluation: Topic Kohärenz während der Session. Flüssigkeit der Interaktion, User Engagement. Qualtitative, weiche Maße. Mit und ohne Tuning gemessen → Ergebnis Module tragen signifikant bei Datenset etwas dünn.

Hello Again…HaoLi et al – Zusatzinformation im Paper

Appendix: Literatur zum long-term Dialog. Wissen extern (common sense) [39,40] intern [2,3]

intern: historical events, personas [9,3,10,43]

Literatur zu LLM basierten autonomen Agenten: im Paper

autonome Wahrnehmung der Umgebung,

Entscheidungsfähigkeit, Problemlösefähigkeit

task-orientiert, simulationsorientiert

Hello Again…HaoLi et al – Zusatzinformation im Paper

task orientiert: web assistance, game playing, software development
simulationsorientiert: emotive, cognitive, behavior,
roleplaying & psychological studies
conflict resolution [45]
recommendors [21,47]
personalization, character [48-50]

Datensets: MLC[1], CC[7]
Personal Chat Dataset [5] → alle Human/Human Crowdworkers
Split in Experimenten z.B. 4000 conversationen training 500/500 test/eval
Ubuntu IRC benchmark set 300000 multiparty

Hello Again…HaoLi et al – Zusatzinformation im Paper

Prompt Persona EXTRACTION S. 16

D.2 Prompt of Persona Extraction

Prompt 2: Persona Extraction Prompt

SYS PROMPT:

You excel at extracting user personal traits from their words, a renowned local communication expert.

USER PROMPT:

If no traits can be extracted in the sentence, you should reply NO_TRAIT. Given you some format examples of traits extraction, such as:
1. No, I have no longer serve in the millitary, I had served up the full term that I signed up for, and now work outside of the millitary.
Extracted Traits: I now work elsewhere. I used to be in the military.
2. That must a been some kind of endeavor. Its great that people are aware of issues that arise in their homes, otherwise it can be very problematic in the future.
NO_TRAIT
Please extract the personal traits who said this sentence (no more than 20 words):{*sentence*}

Hello Again…HaoLi et al – Zusatzinformation im Paper

Prompt Agent RESPONSE S. 17

Prompt 4: Agent Response Generation Prompt

SYS PROMPT:

As a communication expert with outstanding communication habits, you embody the role of {*agent name*} throughout the following dialogues. Here are some of your distinctive personal traits: {*agent traits*}.

USER PROMPT:

<CONTEXT>
Drawing from your recent conversation with {*user name*}:
{*context*}
<MEMORY>
The memories linked to the ongoing conversation are:
{*memories*}
<USER TRAITS> In recent conversations with this user, you've noticed that {*user name*} possesses the following personal traits:
{*user traits*}
Now, please role-play as {*agent name*} to continue the dialogue between {*agent name*} and {*user name*}.
{*user name*} just said: {*input*}
Please respond to {*user name*}'s statement using the following format (maximum **30** words, **must be in English**):
RESPONSE:

Context and time sensitive memory…Alonso et al – Introduction

Toward Conversational Agents with Context and Time Sensitive Long-term Memory

Nick Alonso
Zyphra
nick@zyphra.com

Tomás Figliolia
Zyphra
tom@zyphra.com

Anthony Ndirango
Zyphra
anthony@zyphra.com

Beren Millidge
Zyphra
beren@zyphra.com

Abstract

There has recently been growing interest in conversational agents with long-term memory which has led to the rapid development of language models that use retrieval-augmented generation (RAG). Until recently, most work on RAG has focused on information retrieval from large databases of texts, like Wikipedia, rather than information from long-form conversations. In this paper, we argue that

Context and time sensitive memory…Alonso et al – Introduction

Ausgangsthese: starkes Interesse an Conversational Agents mit longterm memory:

LLM + RAG Imlementierung

Retrieval von M_l hat zwei Eigenarten: zeit/event basierte Anfragen + ambige Anfragen die Kontext brauchen

Problem: Datenset mit Queries beider Eigenarten: Standard RAG Ansatz (--)

Lösungsansatz: Kombination in neuem Framework:

- Chain of Table Search
- Standard Vektordatenbank
- Promt für Disambiguation

Ergebnis: auf speziellem Datenset besser.

Context and time sensitive memory…Alonso et al – Introduction

Problemeröffnung: zwei Problemsettings, die in den üblichen RAG Szenarien nicht abgedeckt werden:

1) Conversational Metadata based Queries: „what was the idea we were working on last time“ gegeben Metadaten soll das Modell Inhalt/Topic liefern.

2) Ambige Fragen, auch Anaphern, Pronomen Resolution: das LLM kann im Textzusammenhang auflösen, das Retrieval nicht.

Es gibt dazu keine Benchmarks, keine annotierten Datensets. [8,9] Retrieving aus der Chathistory ist nicht im Testset abgedeckt. Die annotierten Ambiguitäten liegen in der Query und beziehen sich nicht auf das Retrieval.

Für IR ist Metadatensuche bekannt aber nicht für Conv Agents Retrievalfunktion.

Context and time sensitive memory…Alonso et al – Introduction

Contributions:

1) Datenset aufbereitet für Conv Agents das ermöglicht zu testen wie gut der Agent für ambige und metadaten und kombinierte Probleme funktioniert

2) neues Retrievalmodell das Vektordatenbanken mit einer Chain of Tabels [28] Suche verbindet . Klassifikator ob die Query aof Metadaten, Content oder beides abziehlt.

Context and time sensitive memory…Alonso et al – Related Work

Longterm Dialogue: bis zum Paper oft nur Datensätze mit relativ kurzen Dialogen verfügbar siehe Lit S. 3.

GoodAI Benchmarks [4] mit sehr langer Distanz.

LocoMo Datenset: synthetisch mit GPT4

Temporal Reasoning: kaum etwas verfügbar dass das Retrieval Problem beschreibt

Ambiges Conversational Reasoning: SoA Datensets Benchmarks im Retrieval.

Benutze Fragen wo Objekte explizit benannt sind. Kein Agent Datenset vorhanden.

Table based RAG: strukturierte Anfragen, spezielle Modelle, Chain of Table Modell

Context and time sensitive memory…Alonso et al – Related Work

Chatbots mit Longterm Memory:

Simples Semantisches Retrieval: Literatur S.4

Advanced Techniques [17,19,4] aber begrenzt auf semantisches Retrieval.

Context and time sensitive memory…Alonso et al – Datensets

LOCOMO als Basis 35 Dialoge mit shortterm, longterm.

=> Modifikation. Anfangs nur 12 Dialoge nutzbar. Zu wenig history, zu kurz.

Restliche angereichert.

Fragetypen an das Datenset:

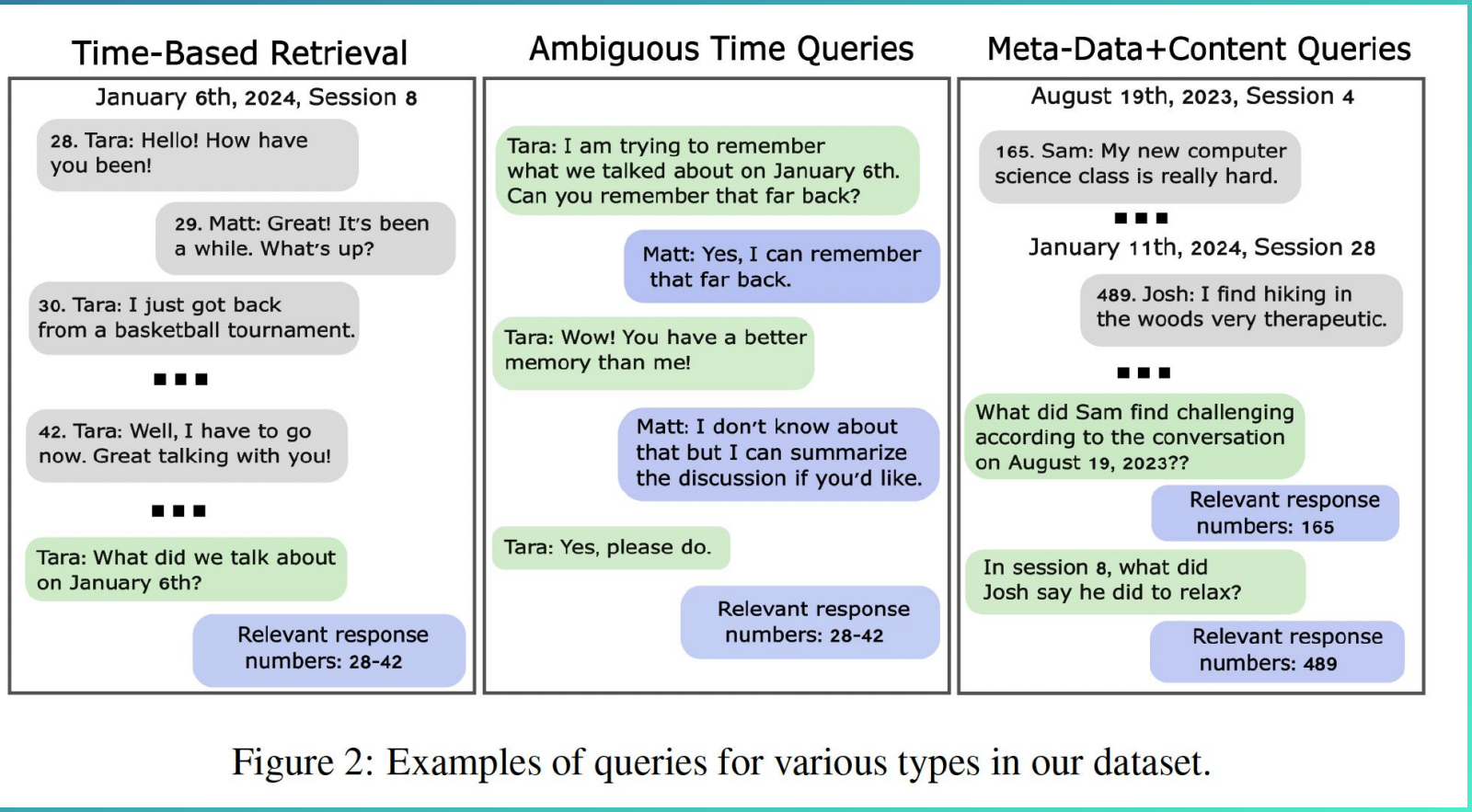


Figure 2: Examples of queries for various types in our dataset.

Alonso et al. S.5

Context and time sensitive memory…Alonso et al – Datensets

Testsetup: Performanz des IR Modell soll separiert vom Generatormodell gemessen werden. Recall ist wichtiger da LLMs irrelevanten Kontext ignorieren können er also später wieder ausgeschlossen werden kann.

Daher F2 Maß.

Time base Queries: 11 Templates im Datenset z.B. „What did we discuss in Session x“ single hop

Ambiguous Timebased Queries: Obfuscication, Pronomen. “When did we discuss this earlier“

Time + Content Q.: Multihop “what video game did Jolene mention playing with her partner on January 27th 2023“

=> brauchen einen Schritt zwischen 27.01.2023 und dem topic Video Game.

Context and time sensitive memory…Alonso et al – Model

Advanced IR System um ambige Metadatenorientierte Queries besser zu beantworten.

Tabular Database: Jede Antwort besteht aus Text und Metadaten. Sprecher, Datum, Zeit, Session ID: natürlich tabulare Datenbankrepräsentation.

Collumn: Daten bestimmten Formats

Line: Informationen für eine spezifische Antwort. Kombination mit Vektordatenbank für Content über einen Contentlink.

Klassifikation des Fragetyps: Semantische, Meta, Kombination. Few Shot LLM wird benutzt um die Query zu klassifizieren.

Context and time sensitive memory…Alonso et al – Model

Chain of Tables für das Metadatenretrieval: Könnten SQL oder Dataframe Languages wie Pandas benutzen.

Chain of Table Algorithmus [28]: kleine Bibliothek von Funktionen, die aus einer Tabelle Informationen ziehen können.

Das LLM ruft eine Kaskade dieser Funktionen auf.

= Sequenz um den Kontext von Multi-Hub Queries zu erzeugen → im Paper wird die Technik auf Metadatenqueries adaptiert.

Context and time sensitive memory…Alonso et al – Model

Zwei Funktionen um Subsets aus der Tabelle zu ziehen.

f_value(column_name[value1, value2,…])

holt alle Zeilen aus der Tabelle die einen der gelisteten Values in der angegebenen Spalte matchen

f_between(collumn_name, [value1, value2])

holt alle Reihen zwischen value1 und value2

Paper: Funktionen können alle Fragen im Testset aus den Meta-Tables beantworten.

Context and time sensitive memory…Alonso et al – Model

Prompts aus der originalen Chain-of-Table Publikation angepasst.

Beispiele im Paper

Appendix 7

```
We have a table that stores a log of responses between two speakers in a
table format.  Information about response speaker names, and dates are
stored in the table, where each row stores information about one response.
There are functions that allow us to isolate rows in the table.  The
columns in the table are:

/*
Response_Index | Session_Index | Speaker | Day_Name | Week | Date | Time
*/

If the table only needs rows that have a certain value in a certain
column to answer the question, we use 'f_value(column_name, [row_value1,
row_value2, ...])'  to select these rows.  For example,

[EXAMPLES]

Here are examples of chaining together multiple function calls.  Each
chain of function calls is ended when the model outputs <END>.  For
example,

[EXAMPLES]

Finish the following function chain.  Do not write any extra text.  Only
output the first argument of the 'f_value' function.

Current Date:[DATE] Current Time:[TIME] Current Session:[SESSION NUM]
Query:  [QUERY]
Function Chain:  [FUNCTION CHAIN SO FAR] -> [OUTPUT]
```

Context and time sensitive memory…Alonso et al – Model

Kombination aus Meta Daten und semantischem Retrieval

0) Query Disambiguation

1) Query Art durch LLM Klassifikator bestimmen

2) Metadaten Retrieval. ChainofTable. Semantischer Inhalt per Indexlink.

3) Top-k semantic retrieval aus den restlichen Dokumenten

Ausgabe nach treshold

Query Rewrite: Adpat Prompt Methode zur Query Disambiguierung [20]. Few Shot: zeigt dem LLM wie es disambiguieren soll. Siehe Appendix Prompts.

Andere Methoden dazu: Fine Tuning

Context and time sensitive memory…Alonso et al – Limitierungen

Modelle schon etwas outdated: Mistral-Hermes-7b und GPT-3.5-turbo.

Neuere LLMs besser?

Embedding Modell relativ klein < 500 mio Parameter

Datenset etwas klein und natürlich sehr Problem fokussiert.

Trotzdem Gute Problemstellung und Interessantes Vorgehen.

Wichige Fähigkeit von ConvAgents die langfristige Konversationen unterhalten wollen: Verständnis zum Zeitbezug, Ambiguität der Query, interessantes Datenset