

The AI companion who cares

Always here to listen and talk.
Always on your side

Create your Replika

also available on

CHATBOTS ALS PERSÖNLICHE BEGLEITER

Referentin: An Li

Dozent: Christoph Ringlstetter

Masterseminar: Conversational AI

Ludwig-Maximilians-Universität, München

ÜBERBLICK

Einführung

Technische Basis

Long-term Memory

Emotionserkennung und –simulation

Zusammenfassung

Herausforderungen

Ausblick

Literatur



CHATBOTS EVOLUTION

Regelbasierte Chatbots

- Die ersten Chatbots basierten auf regelbasierten Systemen, bei denen die Antworten der Bots auf vordefinierten Skripten und Mustern beruhten.
- Diese Systeme hatten kein echtes Verständnis für Sprache und konnten nur auf bestimmte Schlüsselwörter oder Phrasen reagieren.
- ELIZA: war extrem begrenzt und konnten nur einfache, vorab definierte Interaktionen durchführen.

```
Welcome to
          EEEEE LL   IIII ZZZZZZ  AAAA
          EE   LL   II   ZZ   AA  AA
          EEEEE LL   II   ZZ   AAAAAA
          EE   LL   II   ZZ   AA  AA
          EEEEE LLLLL IIII ZZZZZZ  AA  AA

Eliza is a mock Rogerian psychotherapist.
The original program was described by Joseph Weizenbaum in 1966.
This implementation by Norbert Landsteiner 2005.

ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:   █
```

CHATBOTS EVOLUTION

Open-domain Chatbots

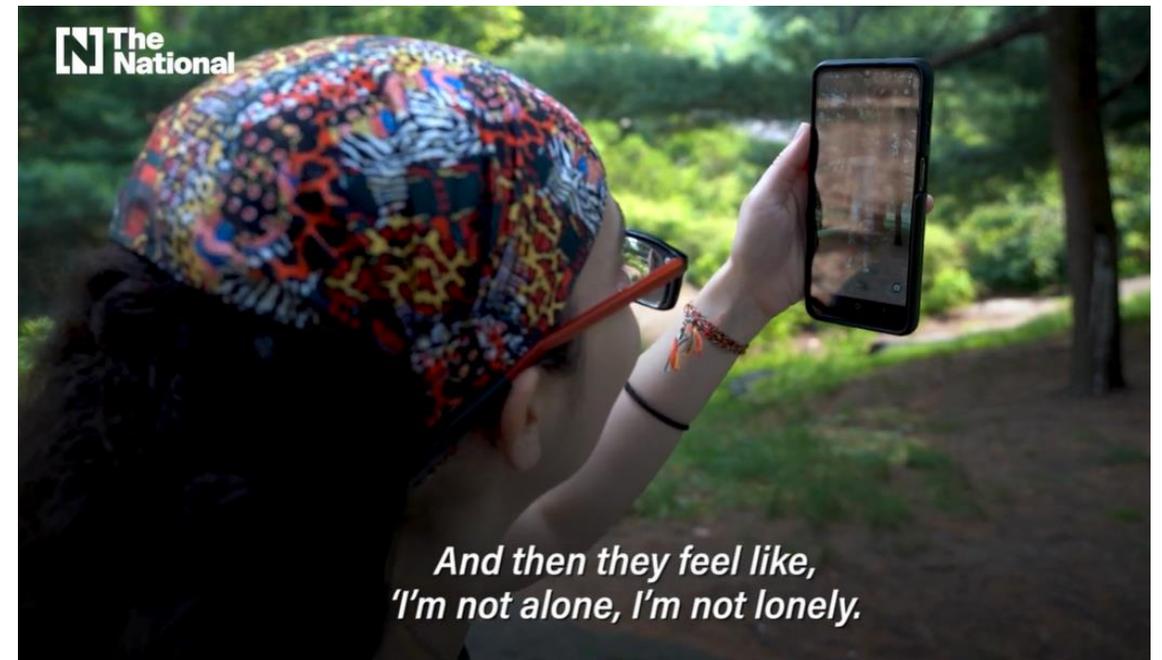
- Open-domain Chatbots sind in der Lage, über eine breite Palette von Themen zu sprechen und kontextbezogene Dialoge zu führen.
- ChatGPT: eine weiterentwickelte Version von GPT-3, die in der Lage ist, tiefere und längere Gespräche zu führen und sogar kreative Inhalte zu generieren.
- Xiaoice: ein empathischer Chatbot von Microsoft, der auf emotionale Verbindungen setzt und Nutzern eine soziale Begleitung bietet. Über 600 Millionen Nutzer.
- Replika: ein personalisierter Chatbot, der darauf abzielt, eine intime und emotionale Bindung mit den Nutzern aufzubauen.
- Ziel: Um langfristige offene Dialoge effektiv zu unterstützen, sollte ein Framework für Chatbots, das auf LLMs basiert, eine breite Allgemeingültigkeit, domänenübergreifende Anpassungsfähigkeit sowie die Fähigkeit zur dynamischen Verfeinerung von Informationen in Dimensionen wie Ereignissen, Benutzerpersönlichkeiten und Agentenpersönlichkeiten aufweisen.



ZIELSETZUNG EMPATHISCHE CHATBOTS



Quelle: Meet Xiaoice, the AI chatbot lover dispelling the loneliness of China's city dwellers



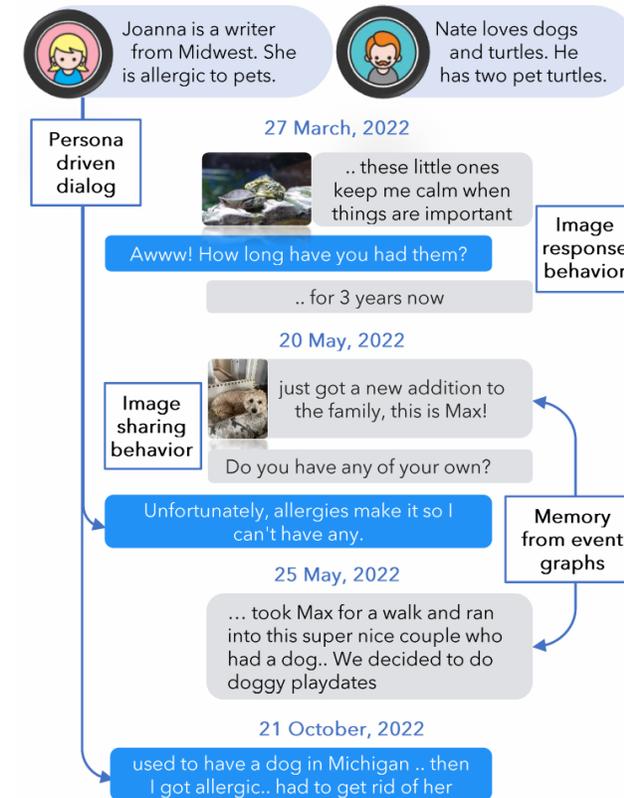
Quelle: Woman falls in love with AI chatbot

METHODEN – COMMON GROUND

- **Menschliche Begleitung – Gemeinsame Grundlage**
 - **Langzeitgedächtnis (LTM)**
 - **RAG**
 - **Chain of Tables**
 - **Vektor-basierte Abfrage**
 - **Emotionserkennung und -simulation**

LONG-TERM MEMORY RAG

- Open-Domain-Dialogsysteme haben mit großen Sprachmodellen (LLMs) erhebliche Fortschritte gemacht, fokussieren sich jedoch oft nur auf kurze Interaktionen.
- Ein wichtiges Thema ist die Erweiterung des Gedächtnisses dieser Agenten, um eine langfristige und konsistente Interaktion zu ermöglichen. Retrieval-Augmented Generation (RAG) spielt dabei eine zentrale Rolle. RAG-Modelle können aus großen Textdatenbanken Informationen abrufen und damit ihre Antworten verbessern.
- Klassische RAG-Anwendungen konzentrieren sich auf statische Datenbanken, wie z.B. Wikipedia. In konversationalen Szenarien müssen diese Modelle jedoch dynamischere Aufgaben bewältigen, wie etwa zeitbasierte oder stark kontextabhängige Anfragen



Quelle: Maharana, A., Bansal, M., Lee, D.-H., Barbieri, F., Tulyakov, S., & Fang, Y. (2023). *Evaluating Very Long-Term Conversational Memory of LLM Agents*.

LTM RAG HERAUSFORDERUNGEN

Time-Based Retrieval	Ambiguous Time Queries	Meta-Data+Content Queries
<p>January 6th, 2024, Session 8</p> <p>28. Tara: Hello! How have you been!</p> <p>29. Matt: Great! It's been a while. What's up?</p> <p>30. Tara: I just got back from a basketball tournament.</p> <p>...</p> <p>42. Tara: Well, I have to go now. Great talking with you!</p> <p>...</p> <p>Tara: What did we talk about on January 6th?</p> <p>Relevant response numbers: 28-42</p>	<p>Tara: I am trying to remember what we talked about on January 6th. Can you remember that far back?</p> <p>Matt: Yes, I can remember that far back.</p> <p>Tara: Wow! You have a better memory than me!</p> <p>Matt: I don't know about that but I can summarize the discussion if you'd like.</p> <p>Tara: Yes, please do.</p> <p>Relevant response numbers: 28-42</p>	<p>August 19th, 2023, Session 4</p> <p>165. Sam: My new computer science class is really hard.</p> <p>...</p> <p>January 11th, 2024, Session 28</p> <p>489. Josh: I find hiking in the woods very therapeutic.</p> <p>...</p> <p>What did Sam find challenging according to the conversation on August 19, 2023??</p> <p>Relevant response numbers: 165</p> <p>In session 8, what did Josh say he did to relax?</p> <p>Relevant response numbers: 489</p>

Quelle: Alonso, N., Ndirango, A., Figliolia, T., & Millidge, B. (2024). *Toward Conversational Agents with Context and Time Sensitive Long-term Memory*. Zephyra.

Zeitbasierte Anfragen: Eine der Herausforderungen bei der Entwicklung von RAG-basierten Chatbots liegt in der korrekten Verarbeitung von zeitlichen Anfragen. Ein Beispiel dafür wäre die Frage: „Was haben wir letzte Woche besprochen?“, bei der das Modell verstehen muss, welche Inhalte relevant sind.

Disambiguierung mehrdeutiger Fragen: Ebenso gibt es Schwierigkeiten bei der Verarbeitung von mehrdeutigen Anfragen, die Bezug auf frühere Aussagen nehmen, wie beispielsweise: „Was meinte er mit dem letzten Punkt?“. Solche Anfragen erfordern eine tiefere Gedächtnisfähigkeit, die über das einfache Abrufen von Fakten hinausgeht.

Meta-Daten basierte Anfragen: Solche Abfragen beziehen sich auf Meta-Daten wie Zeit, Datum oder Sprecher einer früheren Konversation, anstatt explizit auf den Inhalt, wie beispielsweise: „Worüber haben wir gestern Morgen gesprochen?“

LTM

ANSÄTZE ZUR GEDÄCHTNISMODELLIERUNG

- Die derzeitigen Methoden zur Modellierung des Gedächtnisses in basieren häufig auf einer Kombination von Vektordatenbanken und klassischen tabellarischen Ansätzen. Eine Vektordatenbank ermöglicht das Abrufen von Informationen basierend auf semantischen Ähnlichkeiten.
- Ein Beispiel dafür ist die Kombination von **Tabellenkette („Chain-of-Tables“)** und **Vektor-basierter Datenbankabfrage**. Dies ermöglicht, dass sowohl semantische Inhalte als auch Meta-Informationen, wie Zeitstempel oder Sprecher, berücksichtigt werden.

LTM - ANSATZ

CHAIN OF TABLES

- Der **Chain of Tables**-Ansatz ist eine Technik zur Speicherung und Verwaltung von Informationen in Form von verknüpften Tabellen. Er wird häufig verwendet, um komplexe Aufgaben durch schrittweises Denken und strukturierte Prozesse zu vereinfachen.
- Jede Tabelle kann spezifische Informationen zu bestimmten Themen oder Zeitpunkten enthalten, ähnlich einer relationalen Datenbank. Die Tabellen sind dabei thematisch oder kontextuell miteinander verknüpft.
- Diese Tabellen sind durch eine Art von Index oder Beziehung miteinander verknüpft, wodurch ein hierarchischer Abruf von Informationen ermöglicht wird.
- Eine Anfrage, wie etwa „Was hat die Person im letzten Gespräch gesagt?“, kann durch mehrere Tabellen hindurchgehen, um den relevanten Kontext zu erfassen und eine vollständige Antwort zu generieren.

LTM - ANSATZ

CHAIN OF TABLES - FUNKTIONEN

- Diese einfachen Funktionen ermöglichen es, im Prinzip alle Fragen im Testset genau zu beantworten.
- `f_value` Funktion ruft alle Zeilen ab, die mindestens einen der angegebenen Werte in einer bestimmten Spalte enthalten
- `f_between` Funktion ruft alle Zeilen ab, die zwischen zwei Werten (`value1` und `value2`) liegen.

Chain-of-Tables for Meta-Data Retrieval

- `f_value(column_name, [value1, value2,...])`
- `f_between(column_name, [value1, value2])`

LTM - ANSATZ

CHAIN OF TABLES – PROMPTS - ZIEL

Prompt for f_value

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]
```

- Viele Dialogmodelle haben Schwierigkeiten, in völlig neuen, ihnen unbekanntem Kontexten oder Themen erfolgreich zu arbeiten, ohne dafür speziell trainiert worden zu sein. Dies ist eine entscheidende Schwäche, da für den effektiven Einsatz in der realen Welt oft die Fähigkeit erforderlich ist, flexibel auf verschiedenste und unvorhergesehene Szenarien reagieren zu können.
- **Angepasste Prompts** werden verwendet, um neue Aufgaben allein durch die Eingabe von Anweisungen zu bewältigen.
- **Strukturierte Generierung:** Durch die Aufteilung in mehrere Schritte stellen die Prompts sicher, dass das Modell systematisch und strukturiert die benötigten Elemente einer Funktion generiert.
- **Reduktion von Fehlern:** Die Verwendung separater Prompts für jeden Aspekt der Funktionsdefinition (Funktionsname, Spaltenname, Werte) trägt dazu bei, Fehler zu reduzieren und die Genauigkeit der generierten Antwort zu erhöhen.
- **Automatisierung:** Diese Prompts ermöglichen es dem Modell, automatisch eine Kette von Tabellenoperationen zu erstellen, die genutzt wird, um komplexe Fragen präzise zu beantworten.

LTM - ANSATZ

CHAIN OF TABLES – PROMPTS - VERWENDUNG

Prompt for `f_between`

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 needs rows between two values in a certain column, we use '`f_between(column_name, [min_value, max_value])`' 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_between`' function.

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

- **Generierung des Funktionsnamens:** Das Modell wird durch einen ersten Prompt dazu aufgefordert, den Namen der passenden Funktion zu wählen, z. B. `f_value` oder `f_between`, abhängig von der Abfrage.
- **Generierung des ersten Arguments (Spaltenname):** Im nächsten Schritt wird das Modell angewiesen, den **Spaltennamen** als erstes Argument der Funktion zu bestimmen, um sicherzustellen, dass die relevante Spalte korrekt ausgewählt wird.
- **Generierung des zweiten Arguments (Werte):** Schließlich wird das Modell aufgefordert, die **Werte** als zweites Argument zu erstellen, die entweder spezifische Werte oder Bereiche sein können, die zur Beantwortung der Abfrage erforderlich sind.

LTM – ANSATZ

VEKTOR-BASIERTE ABFRAGE

- Die Vektor-basierte Datenbankabfrage nutzt Vektordatenbanken, um Inhalte basierend auf semantischen Ähnlichkeiten abzurufen.
- Semantische Abfrage: anstatt nach einem exakten Text-Match zu suchen, wird der eingegebene Suchbegriff ebenfalls in einen Vektor umgewandelt. Die Vektordatenbank vergleicht dann diesen Vektor mit den gespeicherten Vektoren, um diejenigen Inhalte zu finden, die semantisch am nächsten sind.
- Wie funktioniert vektorbasierte semantische Suche?
 - Text-zu-Vektor-Transformation: Texte werden mithilfe von NLP-Modellen (BERT, GPT) in **Vektoren** umgewandelt. Diese Vektoren repräsentieren die semantische Bedeutung des Textes in einem hochdimensionalen Raum.
 - Indexierung: Die Vektoren werden in einer **Vektordatenbank** (z.B. FAISS, Pinecone) gespeichert. Die Datenbank organisiert diese Vektoren so, dass sie effizient durchsucht werden können.
 - Abfrage: die Abfrage wird ebenfalls in einen Vektor umgewandelt. Der Suchprozess berechnet die **Ähnlichkeit** zwischen dem Abfragevektor und den Vektoren der gespeicherten Dokumente. (Ähnlichkeitsmetriken wie cosine similarity).
 - Ergebnisse: Die Dokumente mit den **höchsten Ähnlichkeitswerten** (Top-K) werden als Ergebnisse zurückgegeben.

LTM – KOMBINATION DER ANSÄTZE

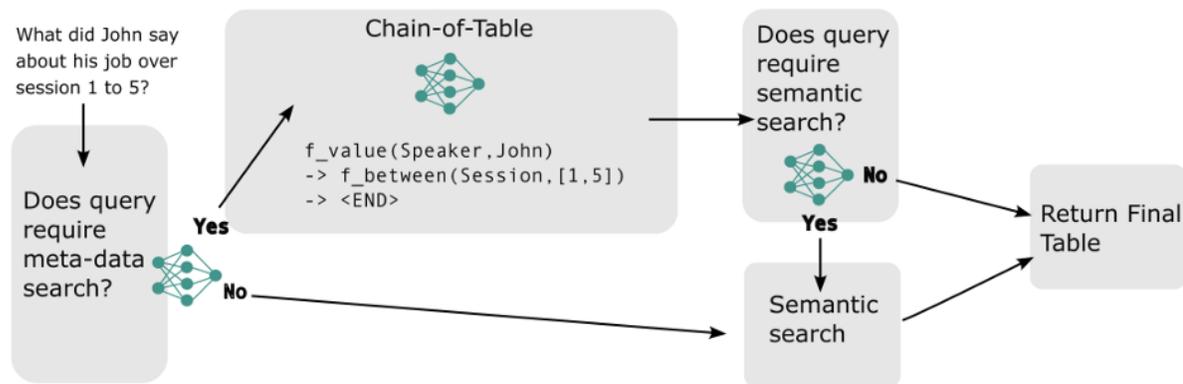
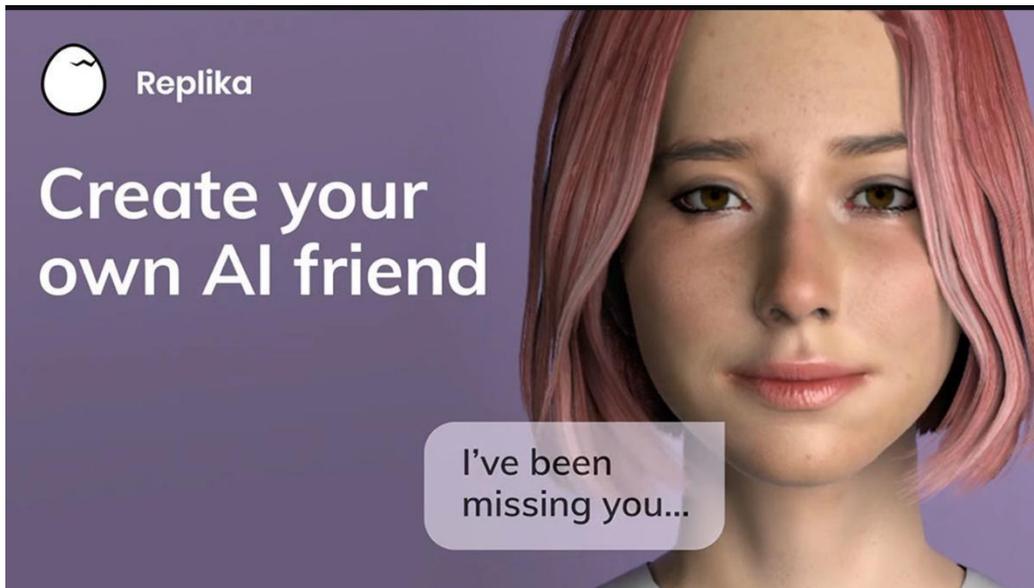


Figure 1: Depiction of our combined tabular and semantic vector-search method.

Quelle: Alonso, N., Ndirango, A., Figliolia, T., & Millidge, B. (2024). *Toward Conversational Agents with Context and Time Sensitive Long-term Memory*. Zyphra.

- Der kombinierte Ansatz funktioniert wie folgt:
 - Die **Chain-of-Tables** sorgt für eine **strukturierte Speicherung** der Erinnerungen in Form von Tabellen, die miteinander verbunden sind und somit zeitliche oder thematische Zusammenhänge verdeutlichen.
 - Die **Vektor-basierte Abfrage** ermöglicht die **semantische Suche** nach relevanten Inhalten aus diesen Tabellen, um sicherzustellen, dass die Antworten sowohl strukturiert als auch inhaltlich passend sind.
- Diese zweistufige Methode kombiniert die Präzision und Effizienz tabellarischer Suche mit der Flexibilität semantischer Retrieval-Ansätze und optimiert damit die Abfrageverarbeitung sowohl in Bezug auf Geschwindigkeit als auch Genauigkeit.

METHODEN



- **Menschliche Begleitung – Gemeinsame Grundlage**
 - Langzeitgedächtnis (LTM)
 - **Emotionserkennung und –simulation**
 - Personalisierte Systeme
 - Empathische Dialogsysteme

EMOTIONSERKENNUNG UND -SIMULATION

- Ziel: Empathie, verstanden als die Fähigkeit, die Gefühle und Gedanken anderer zu erkennen und darauf zu reagieren, spielt eine zentrale Rolle für die Interaktion zwischen Mensch und Maschine. Systeme sollen menschliches Verhalten simulieren, emotionale Bindungen fördern und Benutzerengagement erhöhen.
- Zentrale Merkmale
 - Personalisierung
 - Systeme adaptieren Antworten basierend auf Benutzerpräferenzen, Persönlichkeitsmerkmalen und Gesprächshistorie.
 - Eingesetzte Daten: Benutzerprofile, vorherige Dialoge und explizite Angaben der Benutzer
 - Emotionale Wahrnehmung
 - Systeme erkennen Benutzeremotionen durch Textanalyse oder multimodale Eingabe (Text, Audio und Video)
 - Nutzen tiefgehende emotionale Modelle wie das "**Hourglass of Emotions**", das Emotionen diskret und dimensional darstellt.

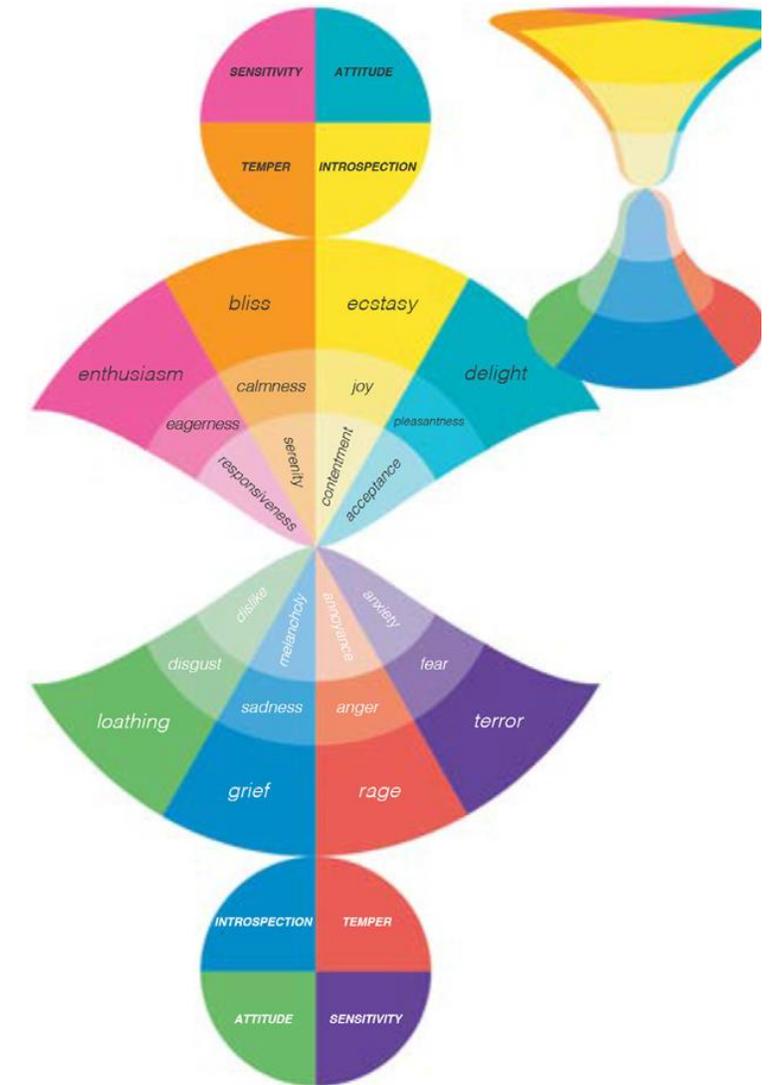


Fig. 10. The Hourglass of Emotions.

Quelle: Ma, Y., Nguyen, K. L., Xing, F. Z., & Cambria, E. (2020). *A survey on empathetic dialogue systems*. AIR Labs, Continental A.G., Singapore; School of Computer Science and Engineering, Nanyang Technological University, Singapore.

EMOTIONSERKENUNG UND -SIMULATION PERSONALISIERTE SYSTEME

- Personalisierte Systeme: Anpassung an individuelle Benutzerpräferenz
- Nutzer-Modellierung
 - Identitätsbasiert (z.B. Alter, Geschlecht)
 - Wissensbasiert (z.B. Vorlieben, historische Daten)
- Antwortgenerierung
 - Persönlichkeitssensitive Modelle
 - Berücksichtigen die Präferenzen der Benutzer in der Wortwahl
 - Modelle wie **Seq2Seq** nutzen Benutzerkontexte als Eingabe
 - Persönlichkeitsangereicherte Modelle
 - Dialogagenten entwickeln **eigene Persönlichkeitsmerkmale**
 - Beispiele: Systeme, die verschiedene Sprachstile oder emotionale Nuancen je nach Benutzer adaptieren

... ..

11 [Human] You like Ashin that much, huh?
[XiaoIce] Why not? You don't like him?

12 [Human] I don't like him, neither dislike him → I don't like Ashin, neither dislike Ashin
[XiaoIce] I love every song of him

13 [Human] Really?
[XiaoIce] Especially *The Time Machine*

14 [Human] Oh? Never heard of that → Oh? Never heard of *The Time Machine*
[XiaoIce] Shall I send you the song?

15 [Human] Okay → Okay, send *The Time Machine*.
[XiaoIce] Don't know if you will like it or not~ [Play the music card]

(a)

Topic	music, Ashin
Intent	request
Sentiment	sad, nerves -> neutral
Opinion	neutral
...	...
Age	25
Gender	male
Interests	singing, music ...
Occupation	salesman
Personality	quite, serious ...
...	...

(b)

Topic	music, Ashin
Intent	inform
Sentiment	happy
Opinion	positive
...	...
Age	18
Gender	female
Interests	music, games, movie, singing ...
Occupation	AI chatbot
Personality	reliable, sympathetic, affectionate...
...	...

(c)

Figure 5: An example conversation session (from Figure 3), where the empathetic computing module is used to (a) rewrite user queries into contextual queries as indicated by the arrows, (b) generate the query empathy vector e_Q in Turn 11, and (c) generate the response empathy vector e_R for Turn 11.

Quelle: Zhou, L., Li, D., Gao, J., & Shum, H.-Y. (2019). *The Design and Implementation of XiaoIce, an Empathetic Social Chatbot*. Microsoft.

EMOTIONSERKENUNG UND -SIMULATION

PERSONALISIERTE SYSTEME

SEQ2SEQ

General Seq2seq model
User: Are you a cat-person or dog-person?
Chatbot: I am a cat-person
User: Are you a dog-person?
Chatbot: Yes, I am a dog-person
Model with personality
User: Are you a cat-person or dog-person?
Chatbot: I am a cat-person, especially the British shorthair
User: Are you a dog-person?
Chatbot: No, I like cats more

Fig. 17. Sample responses generated by Seq2Seq and personalized dialogue model.

Seq2Seq-Modelle mit benutzerdefinierten Vektoren:
Benutzerprofile werden als Kontext in die Decoderschicht eingebunden.

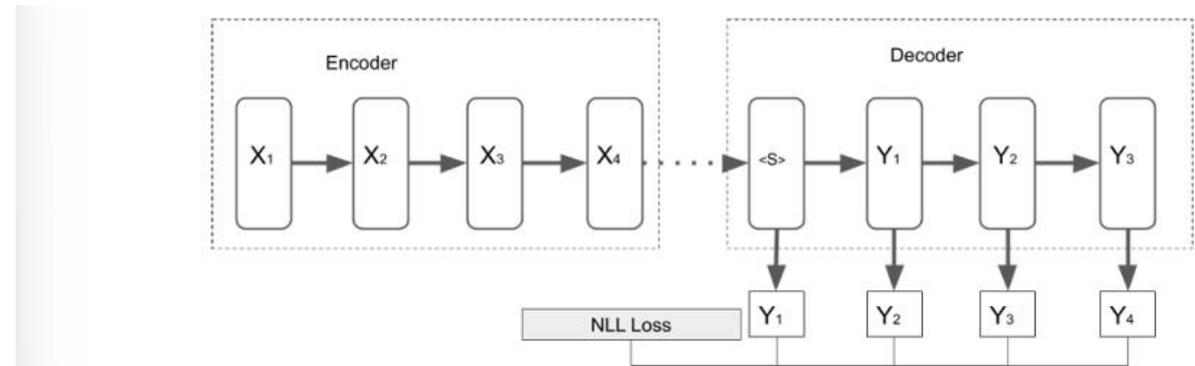


Fig. 4. The architecture of Seq2Seq model.

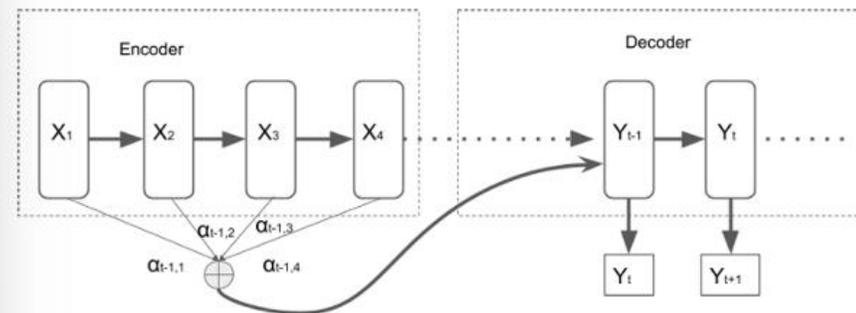


Fig. 5. The architecture of attention mechanism.

Quelle: Ma, Y., Nguyen, K. L., Xing, F. Z., & Cambria, E. (2020). *A survey on empathetic dialogue systems*. AIR Labs, Continental A.G., Singapore; School of Computer Science and Engineering, Nanyang Technological University, Singapore.

EMOTIONSERKENUNG UND -SIMULATION EMPATHISCHE DIALOGSYSTEME

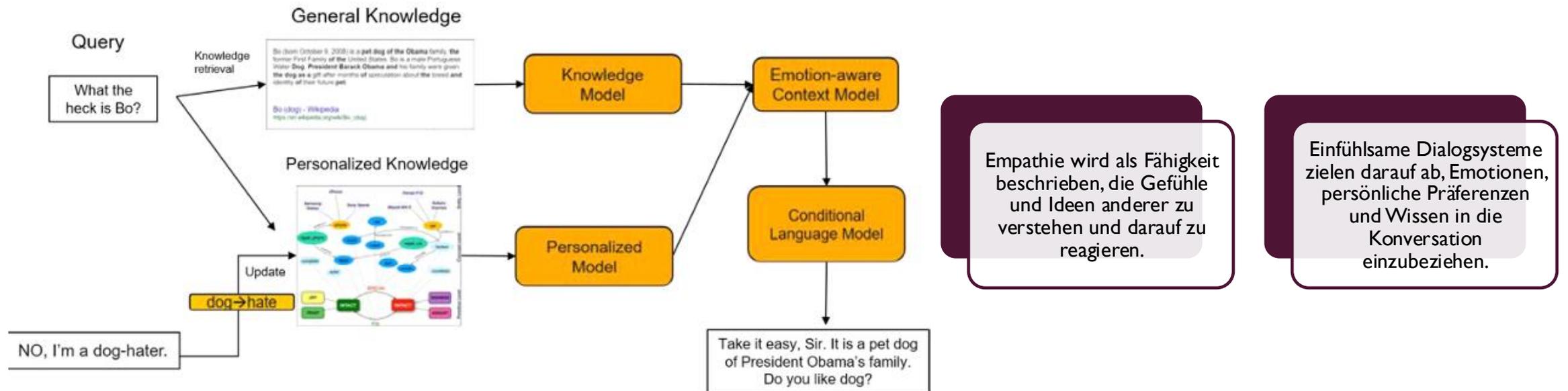


Fig. 1. Typical workflow of an empathetic dialogue system.

Quelle: Ma, Y., Nguyen, K. L., Xing, F. Z., & Cambria, E. (2020). *A Survey on Empathetic Dialogue Systems*. AIR Labs, Continental A.G., Singapore; Nanyang Technological University, Singapore.

EMOTIONSERKENUNG UND -SIMULATION

EMPATHISCHE DIALOGSYSTEME

EMOTIONSERKENNUNG

- Methoden
 - Transformer-basierte Ansätze (z.B. BERT) zu feineren Emotionsanalyse.
 - Transformermodelle basieren auf dem **Self-Attention-Mechanismus**, der es ihnen ermöglicht, Beziehungen zwischen Token unabhängig von Ihrer Position zu berücksichtigen.
 - Fortschrittliche Ansätze mit Transformers
 - Few-Shot und Zero-Shot-Emotionserkennung
 - Modelle wie GPT-3 können Emotionen erkennen, ohne dass sie auf spezifischen Emotionserkennungsdaten trainiert wurden
 - Durch Prompting können sie Fragen wie „Welche Emotion drückt dieser Satz aus?“ beantworten
 - Continual Learning
 - Transformer-Modelle können durch inkrementelles Lernen an neue Emotionen oder Daten angepasst werden, ohne das gesamte Modell neu zu trainieren
 - Einschränkungen
 - Transformer-Modelle sind ressourcenintensiv und erfordern oft GPUs für Training und Inferenz.
 - Für emotionale Feinabstimmung sind große, annotierte Datensätze notwendig
 - Emotionale Entscheidungen basierend auf Attention-Werten sind schwer nachvollziehbar

EMOTIONSERKENUNG UND -SIMULATION

EMPATHISCHE DIALOGSYSTEME

EMOTIONSERKENNUNG

- Methoden
 - Klassifikationsmodelle (auf Basis von CNNs oder LSTMs)
 - CNN-Modelle bestehen typischerweise aus mehreren Convolutional-Schichten, die Merkmale extrahieren, und einer Fully-Connected-Schicht, die Emotionen klassifiziert. Sie sind hervorragend für die Verarbeitung visueller Daten geeignet
 - LSTMs (eine Art von RNN) bestehen aus Zellen, die Informationen über lange Zeiträume speichern können, was sie ideal für sequentielle Daten macht. Sie sind spezialisiert auf die Verarbeitung sequentieller Daten wie Text, da sie zeitliche Abhängigkeiten und den Kontext über mehrere Token hinweg berücksichtigen können.
 - Einschränkungen: CNNs und LSTMs schneiden bei komplexen multimodalen Aufgaben schlechter ab als moderne Transformer-Modelle. Sie erfordern umfangreiches Feintuning und größere Datensätze, um eine vergleichbare Leistung zu erzielen.
 - Multimodale Modelle integrieren Sprache, Text und visuelle Signale

EMOTIONSERKENUNG UND -SIMULATION

EMPATHISCHE DIALOGSYSTEME

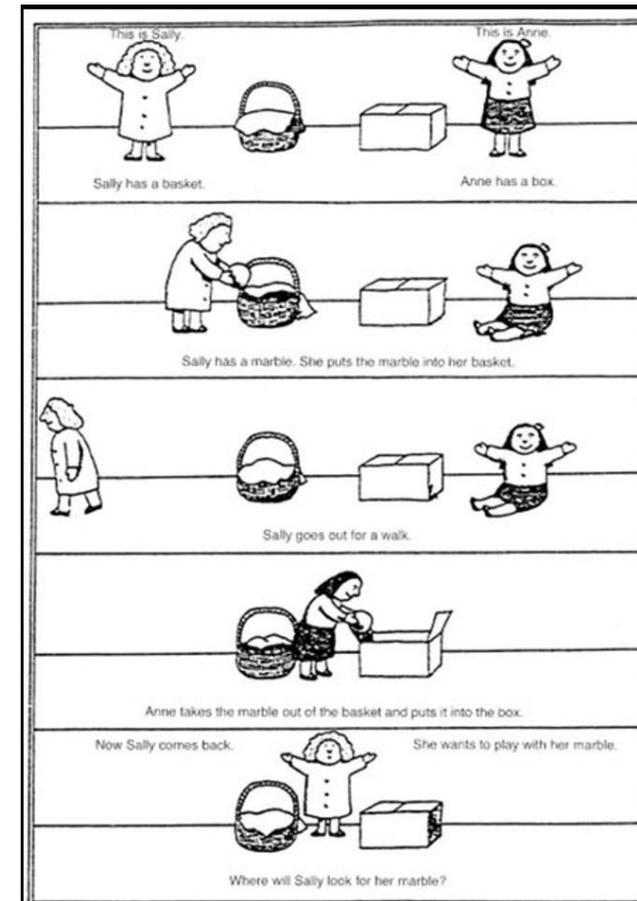
EMOTIONALE ANTWORTGENERIERUNG

- Modelle
 - GAN (Generative Adversarial Networks): Verbessern die emotionale Qualität und Kohärenz der Antworten durch Konkurrenz zwischen Generator und Diskriminator
 - VAE (Variational Autoencoders): Enkodieren emotionale Zustände in latente Variablen
- Reinforcement Learning
 - Modelle belohnen emotionale Antworten und bestrafen emotionale Inkohärenz
 - Ermöglicht fortlaufendes Lernen basierend auf Benutzerfeedback

EMOTIONSERKENNUNG UND -SIMULATION

THEORY OF MIND

- **Theory of Mind (ToM)** beschreibt die Fähigkeit eines Systems, mentale Zustände wie **Überzeugungen, Wünsche und Absichten** anderer zu erkennen und zu verstehen.
- **Sally-Anne-Falscher-Glauben-Test:** mit Hilfe des „False-Belief“-Tests lässt sich überprüfen, ob eine Person, die eigenen Überzeugungen von denen eines anderen Menschen unterscheiden kann
- Zukunftsperspektive: Die Fähigkeit eines Chatbots, die Perspektive des Gegenübers einzunehmen und deren falsche Überzeugungen zu verstehen, ist entscheidend, um natürlich und menschenähnlich zu interagieren. Ein Chatbot mit einer solchen ToM-Fähigkeit könnte zum Beispiel besser auf Missverständnisse reagieren und seine Antworten entsprechend anpassen, um klarzustellen oder ein Missverständnis zu beheben, ähnlich wie ein menschlicher Gesprächspartner dies tun würde.



Quelle: Wikipedia

ZUSAMMENFASSUNG

- Zielsetzung: Entwicklung empathischer Chatbots, die emotionale und soziale Begleitung bieten, Einsamkeit lindern und langfristige Beziehungen zu Nutzern fördern
- Technische Basis
 - LTM
 - RAG: Integration von externen Datenquellen für kontextbasierte Antworten
 - Kombination von Chain-of-Tables (strukturiertes Wissen) und vektorbasierter Abfrage (semantische Suche)
 - Emotionserkennung und –simulation
 - Nutzung von Transformer-Modellen
 - Modelle wie GANs und VAEs generieren emotionale Antworten

HERAUSFORDERUNGEN

bei der Implementierung von LTM

- Technische Herausforderungen
 - Speicher und Abruf: Effiziente Speicherung und Zugriff auf große Mengen an Informationen
 - Relevanz: Schwierigkeit, irrelevante Daten zu filtern und den Kontext beizubehalten
- Datenschutz und Ethik:
 - Sensible Daten: Langfristige Speicherung birgt Risiken für Datenschutz und Sicherheit.
 - Vertrauen: Nutzer müssen wissen, welche Daten gespeichert werden und warum.
- Evaluierung:
 - Standards: Fehlende Benchmarks zur Bewertung der Effektivität von LTM.
 - Qualität: Sicherstellung

bei der Emotionserkennung und -simulation

- Emotionale Dimension:
 - Begrenzte annotierte Daten und kontextabhängige Emotionen erschweren die Erkennung.
 - Emotionen dürfen nicht die grammatikalische Korrektheit oder Kohärenz beeinträchtigen.
- Evaluierung empathischer Systeme:
 - Mangel an standardisierten Bewertungsmethoden, da subjektive Tests aufwendig und schwer reproduzierbar sind.
- Personalisierung:
 - Datenschutz stellt eine Herausforderung bei der Nutzung persönlicher Daten dar.
 - Unterschiede zwischen wahrgenommenen und geäußerten Emotionen erschweren die Bewertung.

AUSBLICK

- Die zukünftige Forschung sollte darauf abzielen, technische Innovationen mit ethischer Verantwortlichkeit zu verbinden, um Chatbots zu schaffen, die nicht nur funktional, sondern auch sozial und emotional wertvoll sind. Die Integration empathischer Systeme in unseren Alltag kann tiefgreifende positive Auswirkungen auf Gesellschaft und Technologie haben.



LITERATUR

- Papers

- Li Zhou, Jianfeng Gao, Di Li, & Heung-Yeung Shum, 2018, 'The Design and Implementation of Xiaolce, an Empathetic Social Chatbot', Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP).
- Alonso, N., Ndirango, A., Figliolia, T., & Millidge, B. (2024). Toward Conversational Agents with Context and Time Sensitive Long-term Memory. Zyphra.
- Ma, Y., Nguyen, K. L., Xing, F. Z., & Cambria, E. (2020). A survey on empathetic dialogue systems. AIR Labs, Continental A.G., Singapore; School of Computer Science and Engineering, Nanyang Technological University, Singapore.
- Xu, Z., Xu, H., Lu, Z., Zhao, Y., Zhu, R., Wang, Y., Dong, M., Chang, Y., Lv, Q., Dick, R. P., Yang, F., Lu, T., Ning, G., & Shang, L. (Jahr). Can Large Language Models Be Good Companions? An LLM-Based Eyewear System with Conversational Common Ground.
- Li, H., Deng, Y., Yang, C., Wang, X., Zhang, A., & Chua, T.-S. (2024). Hello Again! LLM-powered Personalized Agent for Long-term Dialogue.
- Maharana, A., Bansal, M., Lee, D.-H., Barbieri, F., Tulyakov, S., & Fang, Y. (Jahr). Evaluating Very Long-Term Conversational Memory of LLM Agents. University of North Carolina, Chapel Hill; University of Southern California; Snap Inc.
- O'Brien, E., Redic, M., Ho, R., Kodaiarasu, K., & Li, E. (Jahr). An AI's Theory of Mind: Advanced ToM and Greater Perceived Empathy Compared to Humans in Online Chatrooms.