Sprachmodelle

Medienlinguistische Methodik

Arne Rubehn
arne.rubehn@uni-passau.de

13.01.2026

Zusammenfassung

In dieser Sitzung beschäftigen wir uns mit der Funktionsweise von Sprachmodellen, insbesondere von LLM-basierten Chatbots wie ChatGPT. Hierbei wollen wir uns einen Überblick über die zugrundeliegende Technik verschaffen, den Trainingsprozess näher anschauen und die Fähigkeiten und Einschränkungen moderner Chatbots kritisch diskutieren.

Credits

Diese Sitzung orientiert sich stark am (sehr empfehlenswerten) Buch *Weiß die KI, dass sie nichts weiß?* von Katharina Zweig (Zweig, 2025).

1 Einleitung

1.1 Was ist KI?

Das Thema **Künstliche Intelligenz (KI)** ist derzeit in aller Munde, insbesondere als Folge technischer Durchbrüche im Bereich der Sprachtechnologie, die in sehr mächtigen **großen Sprachmodellen** (*large language models*, kurz LLM) gipfelte. Durch diese Assoziation wird KI häufig synonym mit LLMs verwendet; ferner bezeichnet der Begriff sowohl die Forschungsrichtung, die zugrundeliegende Technik, als auch die resultierenden Modelle selbst. Da bietet sich doch zum Einstieg die Frage an: Was ist denn KI jetzt überhaupt?

Tatsächlich gibt es gar keine eindeutige Antwort auf diese so einfach anmutende Frage. Während wir Chatbots zweifelsohne als KI bezeichnen würden, sind zum Beispiel sprachgesteuerte Assistenten (wie Siri oder Alexa) schon so sehr in unseren Alltag integriert, dass wir sie vielleicht nicht mehr unbedingt diesem Label zuordnen würden. Taschenrechner würde heutzutage wohl kaum noch jemand als KI bezeichnen; auch Schachroboter waren schon in den 90er-Jahren auf einem Niveau, dass sie Schachgroßmeister besiegen konnten. Als die jeweiligen Systeme brandneu waren, ist der Begriff KI allerdings durchaus angemessen gewesen (und wurde auch entsprechend verwendet). Wir sehen also, dass wir allgemein neue Technologien als KI bezeichnen, von denen wir zuvor nicht geglaubt haben, dass sie technisch umsetzbar ist. Der Begriff der Künstlichen Intelligenz ist also ein **bewegliches Ziel**! (Zweig, 2025, § 3)

In der Tat gibt es auch von der technischen Seite her keine saubere Definition von KI – ob wir Systeme als „intelligent“ wahrnehmen oder nicht ist abhängig von ihrem Output, nicht von der zugrundeliegenden Technologie. Es handlet sich also stets um ein subjektives, menschliches Urteil, nicht um einen feststehenden technischen Begriff. Marvin Minsky,

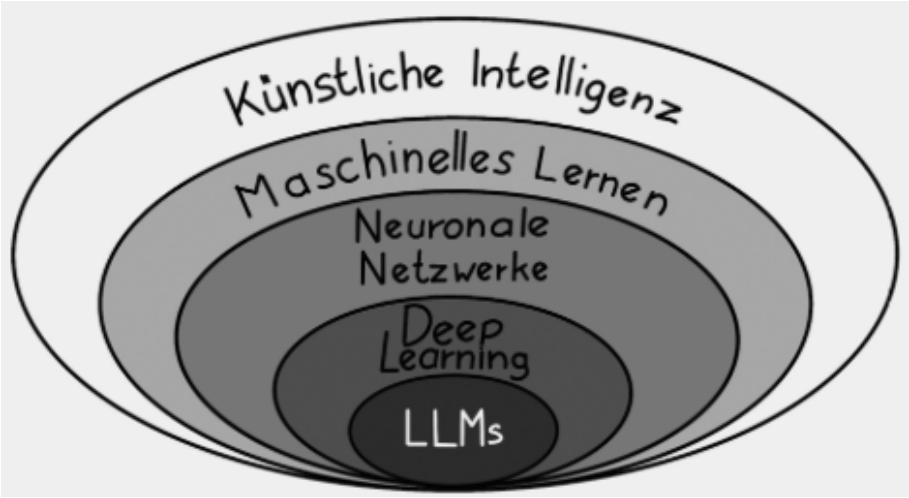

Abbildung 1: Hierarchische Einordnung verschiedener Begriffe um den Themenkomplex KI (Zweig, 2025).

einer der Urväter der KI-Forschung, beschreibt KI als „die Wissenschaft, die Methoden entwickelt, mit denen Computer Aufgaben erledigen können, für die ein Mensch Intelligenz benötigt“ (Minsky, 1968; übersetzt von und zitiert aus Zweig, 2025, § 3). Das würde ja auch auf Taschenrechner zutreffen – nur ist diese Technologie inzwischen so alt und alltäglich, dass es für uns nicht mehr bemerkenswert ist, dass die Maschine das kann.

Wir möchten uns dem Begriff der KI hin zu modernen LLMs mit einem kleinen Exkurs in die Vergangenheit annähern. Während KI heutzutage meistens Modelle meint, die dem Paradigma des **Maschinellen Lernens** (*machine learning*) folgen, hat die Sprachtechnologie lange Zeit vor Allem **regelbasierte Ansätze** entwickelt. Eine Abgrenzung verschiedener Begriffe, die häufig in diesem Themenkomplex verwendet werden, ist in Abb. 1 visualisiert. Ein Beispiel für regelbasierte Ansätze hingegen ist das System **ELIZA**, das wir uns nun im Näheren ansehen.

1.2 ELIZA

Wir lassen den Ansatz des maschinellen Lernens für einen Moment links liegen und widmen uns einem regelbasierten Ansatz. Solche Ansätze basieren auf einem Regelwerk, das der Maschine genaue Anweisungen gibt, was sie wann zu tun und zu lassen hat. Diese Regeln werden von Menschen programmiert und von der Maschine genau befolgt – daher sind resultierende Systeme **deterministisch**, das heißt, dass derselbe Input immer zum selben Output führt.

Das System ELIZA (Weizenbaum, 1966) wurde in den 60er-Jahren von Joseph Weizenbaum entwickelt und wird häufig als erster Chatbot der Geschichte beschrieben. Eine Implementierung von ELIZA findet sich heute noch auf der Webseite des nach ihm be-

nannten Weizenbaum-Instituts.[1]

Wenn man sich ein wenig mit ELIZA „unterhält“, fällt einem auf, dass das System (für heutige Standards) doch ziemlich primitiv ist. Einem von Hand erstellten Regelwerk folgend erkennt die Maschine gewisse Stichwörter und Muster, die sie dann in Rückfragen oder Aufforderungen umformuliert. Man merkt schnell: Wirklich neuer Inhalt kommt von ELIZA nicht, es werden lediglich syntaktische Operationen zur Umstellung der User-Eingabe befolgt.

Viel interessanter als die technische Funktionsweise von ELIZA selbst finde ich allerdings die Reaktion der Menschen darauf. Die Leute, die ELIZA ausprobieren durften, waren hellauf begeistert von den Fähigkeiten des Systems – sie bauten persönliche Beziehungen auf, fühlten sich verstanden und teilten sogar so intime Details, dass sie ihre Chatverläufe nicht mit Weizenbaum teilen wollten! Das Alles geschah in dem Wissen, dass die Menschen mit einer Maschine sprachen, die handgeschriebene Regeln befolgte – und entsprechend keinerlei eigene kommunikative Intentionen, oder gar Intelligenz besaß. Weizenbaum zeigte sich durchaus schockiert von dieser Nutzung seines Systems – so lächerlich dieses Verhalten aus heutiger Perspektive scheinen mag, könnte es ein sehr aufschlussreicher Denkanstoß für die Bewertung heutiger Systeme sein.

1.3 Maschinelles Lernen

Moderne Sprachmodelle basieren nicht auf strikten Regeln, sondern auf gelernten **statistischen Mustern** – menschliche Sprache ist scheinbar zu komplex, um sie anhand von klaren Regeln zu modellieren. **Maschinelles Lernen** bezeichnet eine Familie von Methoden, durch die Modelle statistische Zusammenhänge aus Trainingsdaten lernen können. Das geschieht dadurch, dass eine **Kostenfunktion** minimiert wird – sozusagen eine mathematische Formel, die berechnet, wie weit die Vorhersagen des Modells von den tatsächlich beobachteten Daten abweichen.

Die Outputs solcher Modelle basieren also nicht auf klaren Regeln (wenn X, dann Y), sondern auf gelernten Wahrscheinlichkeiten ($\approx$ X und Y treten häufig gemeinsam auf). Der Trainingsprozess ist **nicht deterministisch** – selbst formell identische Modelle können auf identischen Daten teils andere Muster lernen.

[1] `https://jw.weizenbaum-institut.de/` (aufgerufen am 13.01.2026)

2 Funktionsweise von Sprachmodellen

2.1 Texterzeugung

Moderne Sprachmodelle sind statistische Modelle, die im Kern lediglich lernen, welche **Wortfolgen** wahrscheinlich sind, und welche nicht. Dass sie nicht deterministisch sind, kann man auch an einfachen Experimenten sehen – beginnt man drei separate Chats mit dem Chatbot seines Vertrauens und stellt drei Mal die selbe Frage, werden sich die Antworten in allen drei Fällen vermutlich leicht voneinander unterscheiden. Dieses Verhalten ist durchaus so gewünscht – es bringt Varianz in die Antworten, wodurch sie weniger mechanisch (und daher menschlicher) wirken.

Doch wie erzeugen die Sprachmodelle (die ja das Herz eines jeden Chatbots sind) nun Texte? Die Antwort ist erschreckend simpel: Sie generieren Schritt für Schritt das Wort, das basierend auf dem bisherigen Kontext am wahrscheinlichsten ist. Kontext meint hierbei in erster Instanz den Prompt; jedes daraufhin erzeugte Wort wird dem Prompt angehängt und für die Generierung nachfolgender Wörter berücksichtigt. Der Kontext ist also vereinfacht gesagt der gesamte Chatverlauf – und zwar von beiden Seiten, sowohl die Texte des Users, als auch die des Chatbots.[2]

Nach jedem erzeugten Wort geht das Spiel von vorne los: Das zuletzt erzeugte Wort ist nun das letzte Wort des neuen Kontexts, nun wird aus diesem Kontext eine Wahrscheinlichkeitsverteilung über mögliche Folgewörter berechnet, aus denen das wahrscheinlichste[3] ausgewählt wird. That's it – das ist die ganze Magie.

Klingt zu einfach? Jein. Natürlich gibt es einige Details, die beachtet werden müssen, damit das so funktioniert – die schauen wir uns im Folgenden an. Dennoch lässt sich die Texterzeugung von Sprachmodellen auf eine (sehr komplexe) Wahrscheinlichkeitsrechnung herunterbrechen, welches Wort unmittelbar auf den Kontext folgt. Allerdings muss ein Modell ja erstmal lernen, was wahrscheinlich ist und was nicht – wie das funktioniert, schauen wir uns im nächsten Abschnitt an.

2.2 Pretraining

Eine entscheidende Stärke großer Sprachmodelle ist die, dass die Trainingsprozedur in zwei Schritte unterteilt wird. Im ersten Schritt werden hierbei generelle sprachliche Strukturen gelernt, man spricht hierbei vom **Pretraining**. In einem zweiten Schritt werden die

[2] Es gibt immer eine technische Limitierung, wie groß der Kontext sein kann; also wie weit sich ein Sprachmodell bei der Generierung zurückbeziehen kann. LLMs sind aber inzwischen so groß, dass diese Limitierung bei der Diskussion praktisch keine Rolle spielt.

[3] Auch das ist eine leichte Vereinfachung; teilweise wird auch ein unwahrscheinlicheres Wort gewählt, um die Texterzeugung eben variabler zu machen.

Modelle dann erst für ihre spezifischen Aufgaben trainiert – dieses sogenannte **Finetuning** sehen wir uns etwas später an.

Zum Pretraining sind enorme Textmengen erforderlich. Große KI-Konzerne machen ihre Trainingsdaten und -prozeduren nicht mehr öffentlich, aber es ist davon auszugehen, dass heutige LLMs für ihr Pretraining große Teile des Internets „verschlungen“ haben – eine unvorstellbar große Menge an Texten. Der grundlegende Trainingsmechanismus ist aber wieder relativ simpel: Man nehme einen Satz und schneide ihn irgendwo ab. Die Maschine muss dann vorhersagen, welches Wort denn als Nächstes folgt. Liegt die Maschine falsch, entstehen „Kosten“ (anhand der bereits erwähnten Kostenfunktion) – diese gilt es zu minimieren. Wie in anderen Bereichen des maschinellen Lernens geht es mathematisch nur darum, diese Kosten zu minimieren. Das passiert eben dann, wenn Texte erzeugt werden, die syntaktisch und semantisch kohärent sind.

Nehmen wir mal als Beispiel den Satzbeginn *„Ich gehe heute ins …“*. Plausible Fortsetzungen wären Wörter wie *Kino, Theater* oder *Schwimmbad* – Wörter wie *Informatik* oder *verorten* wären hingegen denkbar unplausibel. Aus den enormen Mengen an Trainingsdaten ist es schlussendlich möglich, dass genau diese Muster gelernt und wiedergegeben werden.

2.3 Word Embeddings

Eine der größten Errungenschaften der Sprachtechnologie der letzten Jahrzehnte ist die Entwicklung von **Word Embeddings** (auf Deutsch auch *Worteinbettungen*). Die Intuition ist hierbei, dass Wörter in einem (hochdimensionalen) Raum so angeordnet werden, dass ähnliche Wörter nah beieinander sind (und unähnliche Wörter weit voneinander entfernt). Ähnlichkeit kann sich hierbei sowohl auf die Funktion (Syntax) als auch auf die Bedeutung (Semantik) beziehen.

Von dieser Komponente profitieren auch LLMs ungemein – ein prominentes Merkmal unserer Sprache ist es ja gerade, dass wir Wörter in beliebig viele sinnvolle Sätze neu anordnen können. Ein Modell, das lediglich gesehene Sätze komplettieren kann, ist daher nicht wirklich brauchbar – es bedarf also eines Modelles, das Relationen zwischen einzelnen Wörtern herstellt. So ist es erst möglich, dass Modelle ähnliche Wörter durcheinander ersetzen können und so neue, sinnvolle Kombinationen erzeugen können.

Word Embeddings werden anhand des **Kontextes** gelernt, in denen ein jeweiliges Wort vorkommt. Die Intuition ist hierbei, dass ähnliche Wörter in ähnlichen Kontexten vorkommen, also von ähnlichen Wörtern umgeben werden (diese Idee wurde schon vor der technischen Anwendung als *distributionelle Semantik* bekannt; vgl. Firth, 1957; Harris, 1954). Es stellte sich heraus, dass diese Strategie, Bedeutung mathematisch zu modellieren, nicht nur erstaunlich gut funktioniert, sondern tatsächlich auch erstaunliche Regelmäßigkeiten

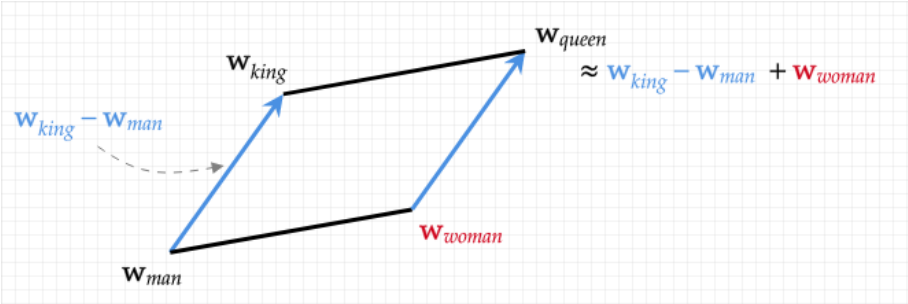

Abbildung 2: Beispiel für analoge Repräsentationen in Word Embeddings

aufweist: So ist zum Beispiel der Vektor (≈ der Weg von einem Punkt zu einem anderen in einem hochdimensionalen Raum) von einem Land zu seiner Hauptstadt in fast allen Fällen nahezu identisch (Mikolov, Yih & Zweig, 2013). Auch Analogien lassen sich auf erstaunlich regelmäßige Weise berechnen: Der Vektor zwischen *man* und *king* ist nahezu identisch mit dem Vektor zwischen *woman* und *queen*; selbiges gilt für auch für die Vektoren *king - queen* und *man - woman* – die Repräsentation für *queen* lässt sich also aus den Repräsentationen für *man, king* und *woman* errechnen (siehe Abb. 2)!

2.4 Attention is all you need

Dieser Abschnitt ist benannt nach dem Artikel, der die Transformer-Architektur vorstellt, auf der alle modernen Sprachmodelle aufbauen (Vaswani et al., 2017). Hier wird klar: Der **Attention Mechanism** (*Aufmerksamkeitsmechanismus*) spielt eine zentrale Rolle. Die Idee ist dabei ziemlich intuitiv: Wenn das Modell das nächste Wort voraussagen soll, sind nicht alle Wörter aus dem bisherigen Kontext gleich relevant. Durch **Attention-Blöcke** lernt das Modell, welche bisherigen Wörter für die Generierung des nächsten Wortes relevant sind. Die **Repräsentation** der einzelnen, gesehenen Wörtern wird also **dynamisch** – je nach Kontext, in dem ein Wort im spezifischen Fall steht, wird dessen Repräsentation angepasst. Während in statischen Word Embeddings für jedes Wort genau eine Repräsentation gelernt wird, erlaubt es der Attention Mechanism, unterschiedliche Repräsentationen für ein Wort basierend auf dem Kontext zu erzeugen, wie in Abb. 3 illustriert.

2.5 Tokenisierung

Wir haben gelernt, dass sich Sprachmodelle im Grunde genommen eine kontextbasierte **Wahrscheinlichkeitsverteilung** über alle Wörter errechnet. Das impliziert ja, dass es eine endliche, abzählbare Menge an Wörtern gibt. Wie ist es dann also möglich, dass Sprachmodelle auch neue, ungesehene Wörter generieren?

Hier habe ich bislang eine Vereinfachung vorgenommen. Die Grundbausteine von Sprachmodellen sind nicht Wörter, sondern **Tokens**. Während häufige Wörter zwar typischer-

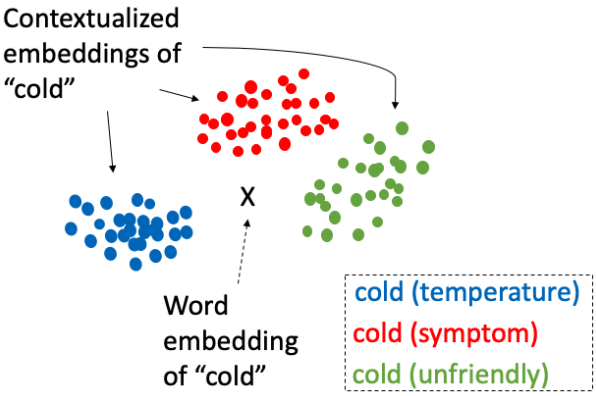

Abbildung 3: Disambiguierung von verschiedenen Wortbedeutungen mit kontextualisierten Repräsentationen.

weise auch einzelnen Tokens entsprechen, können Tokens auch Bruchteile eines Wortes (*Subwords*) oder sogar einzelne Buchstaben sein. Alle bisher genannten Mechanismen assoziieren also **Tokens** miteinander, nicht Wörter. Diese Tokens haben eben den entscheidenden Vorteil, dass sie eine endliche und abzählbare Menge sind (wodurch eine Wahrscheinlichkeitsrechnung möglich wird) – sie können aber beliebig in längere Wörter kombiniert werden; auch solche, die in den Trainingsdaten nie gesehen wurden!

Wie Sprachmodelle Texte in einzelne Tokens unterteilen, kann man sich auf interaktiven Webseiten ansehen – ein Beispiel hierfür wäre `https://gpt-tokenizer.dev`. Eine Webseite, die das technische Innenleben eines Sprachmodelles (GPT-2, eine relativ frühe Version von GPT) schön visualisiert, ist `https://poloclub.github.io/transformer-explainer/`.

2.6 Finetuning

Ein Sprachmodell alleine macht keinen Chatbot. Wie vorhin bereits erwähnt, entstehen „rohe“ Sprachmodelle bereits durch das Pretraining – ein Chatbot ist allerdings ein konkreter Anwendungsfall, für den nochmal eine Feinjustierung (**Finetuning**) in einer zweiten Trainingsstufe notwendig ist. Neben der grundlegenden Sprachfertigkeit muss ein Chatbot ja zum Beispiel auch lernen, wie ein Dialog strukturiert ist und wie er sich zu „benehmen“ hat.

Im allgemeinen Sinne bezeichnet Finetuning einen zweiten Trainingsschritt, bei dem grundlegende Sprachmodelle für einen speziellen Anwendungsfall angepasst werden. Hierbei werden nochmal spezielle, anwendungsspezifische Trainingsdaten verwendet – allerdings sind deutlich kleinere Datenmengen nötig als im Pretraining.

Für Chatbots konkret hat sich eine besondere Form des Finetunings als besonders nützlich erwiesen. Diese lässt sich grob in zwei Stufen unterteilen: Zunächst einmal wird das

Modell auf spezifischen Trainingsdaten in Dialogform trainiert. Diese Art des Trainings entspricht einer sehr klassischen Finetuning-Strategie. Der zweite Schritt ist bemerkenswerter: Hier kommen tatsächlich menschliche Trainer ins Spiel. Das Modell lernt direkt aus menschlichem Feedback – so werden beispielsweise mehrere mögliche Antworten auf einen Prompt präsentiert, und das Modell muss sich für die beste Antwort „entscheiden". Der Mensch bewertet, ob das eine gute Wahl war – entsprechend wird das Modell mathematisch „belohnt" oder „bestraft". Dieses Paradigma nennt sich **Reinforcement Learning**.

3 Fähigkeiten und Einschränkungen

3.1 Was wissen Sprachmodelle?

Wir haben gesehen, dass die grundlegenden Mechanismen von Sprachmodellen im Prinzip relativ simpel sind – die Modelle sind vor Allem gut darin, aus enormen Datenmengen hochkomplexe **Assoziationen** zwischen Tokens zu lernen und diese wiederzugeben. Durch die schiere Menge der Trainingsdaten und die Größe der Modelle ist es möglich, dass Chatbots auf die meisten Fragen tatsächlich auch passende Antworten finden.

Das macht den Anschein, als hätten Chatbots ein tatsächliches Faktenwissen über die Welt. Das ist aber nicht der Fall: Jede Antwort, die generiert wird, ist das Resultat einer Wahrscheinlichkeitsrechnung! Es gibt keine Wissensdatenbank oder dergleichen, auf die Sprachmodelle zugreifen könnten. Alles Weltwissen, das Chatbots wiedergeben, wird implizit durch das Training gelernt. Basierend darauf werden gelernte Muster wiedergegeben, die zwar häufig zu einer logischen und korrekten Aussage führen – allerdings ist es technisch unmöglich, die Wahrheit einer Aussage (im Bezug auf die echte Welt) zu überprüfen! Tatsächlich können Modelle nicht wissen, was sie nicht wissen – sie tun das, was sie immer tun, und erzeugen munter Token für Token. Dabei können zuweilen auch „Fakten" mit viel Selbstbewusstsein erfunden werden – das nennt man meistens **Halluzinationen**.[4]

Ein amüsantes Beispiel dafür lieferte mir neulich erst die Meta AI, die man inzwischen in WhatsApp-Chats anpingen kann. Ein Freund bat die Meta AI um Hilfe bei der Zusammenstellung eines Teams für den Fantasy Manager „Kickbase". Natürlich schien das Modell sehr gewillt, zu kooperieren, und schlug direkt einige Spieler vor – allerdings waren diese Vorschläge absolut unbrauchbar, da die meisten der vorgeschlagenen Spieler nicht mehr bei den genannten Vereinen spielen. Tatsächlich waren die genannten Zuordnungen der Spieler zu ihren angeblichen Vereinen sehr aufschlussreich über das Alter der Trainingsdaten, da die Spieler in aller Regel mit jeweils dem Verein genannt wurden, für den sie vor

[4] Der Begriff Halluzinationen wird viel diskutiert; viele Stimmen halten ihn für nicht besonders geeignet. Da es aber der häufigste Begriff für dieses Phänomen ist, benutze ich ihn hier.

ca. 3 Jahren aufliefen.

Viel interessanter war aber das Phänomen, dass der Chatbot nach ein paar Korrekturen immer stärker anfing, zu halluzinieren: Zunächst wurden Spieler Vereinen zugeordnet, für die sie nie gespielt hatten, und irgendwann erfand der Chatbot einen Spieler namens Matthieu de Ronde, den es nie gegeben hat!

3.2 Stützräder für Sprachmodelle

Dass Sprachmodelle faktisches Wissen nicht direkt abrufen können, ist ein Problem, das KI-Entwickler:innen schnell aufgefallen ist. Moderne Systeme behelfen sich daher mit einem vorgelagerten Modell, das bei einer entsprechenden Anfrage eine **Internetsuche** durchführt und relevante Texte extrahiert. Diese werden dann – unsichtbar für den User – an den **Kontext angehängt**, sodass sich das Sprachmodell bei der Textgenerierung direkt darauf beziehen kann. Dieser Mechanismus nennt sich **Retrieval-Augmented Generation** und ist einer von mehreren Behilfsmechanismen, die sich KI-Entwickler:innen überlegt haben, um die inhärenten Schwächen eines generativen Sprachmodelles abzufangen.

Ein weiteres Beispiel hierfür ist das „Erdbeerproblem“ – die Beobachtung, dass Sprachmodelle nicht zählen können, wie oft der Buchstabe R in *strawberry* oder *Erdbeere* vorkommt, verbreitete sich schnell im Internet. Wie kann so eine scheinbar einfache Frage dem Sprachmodell solche Schwierigkeiten bereiten?

Die Antwort liegt auch wieder in der technischen Funktionsweise des Modells. Wir haben bereits besprochen, dass die kleinste Recheneinheit eines Sprachmodelles die **Token** sind, zwischen denen alle möglichen Beziehungen gelernt werden. Das Wort *Erdbeere* zum Beispiel ist unterteilt in vier Tokens: `E-rd-be-ere`. Jedes Token wird innerhalb des Systems lediglich als eine ganze Zahl dargestellt – für GPT-5 hat das Wort Erdbeere also die Form `36 9290 1464 512`. Sprachmodelle „sehen“ also lediglich eine Kette von Zahlen, die erst im Nachhinein in die entsprechenden Schriftzeichen zurückübersetzt werden!

Fragt man jedoch ChatGPT-5.2 (die beim Verfassen dieses Handouts aktuellste Version), scheint es in der Lage zu sein, Buchstaben in Wörtern korrekt zu zählen – das gilt nicht nur für häufige Wörter wie *Erdbeere*, sondern auch für Pseudowörter oder sogar komplett zufällige Buchstabenfolgen. Kann die KI nun also doch in die Tokens hineinsehen? Nein, auch hier haben sich die Entwickler:innen einen Kniff überlegt, um ihrem Chatbot zu helfen: Für eine spezielle Familie von Problemen gibt es klare Anweisungen, die sich einfach in Programmiercode übersetzen lassen. Wieder ist ein System vorgelagert, das solche Fragestellungen erkennt und einfache Python-Skripts ausführt – deren Output wird dann wieder versteckt an den Kontext angehängt, sodass das Sprachmodell die erfragte Information bereits explizit vorliegen hat.

Dass es immer schwieriger wird, Beispiele zu finden, bei denen fehlerhafte Antworten generiert werden, ist für mich aber kein Beleg dafür, dass die KI „intelligenter“ wird – viel mehr sind es die Entwickler:innen, die mit Hochdruck daran arbeiten, die Schwachstellen ihrer Sprachmodelle durch solche Behilfskonstrukte auszumerzen. Es scheint ein generelles Muster zu sein, in diesen Fällen die relevante Informationen anderswo zu erzeugen und dem Kontext versteckt anzuhängen, damit das Sprachmodell sie in seiner Antwort „wiederholen“ kann.

3.3 Plappernde Papageien oder Superintelligenz?

Abschließend möchte ich noch die offensichtliche und extrem kontroverse Diskussion anstoßen, ob man in die Outputs heutiger KI-Tools tatsächlich irgendeine Form von Intelligenz oder Nachdenken hineininterpretieren kann. Im öffentlichen Diskurs ist hierbei das gesamte Meinungsspektrum vertreten – von plappernden Papageien (*stochastic parrots*; Bender, Gebru, McMillan-Major & Shmitchell, 2021), die keine Ahnung haben, was sie da tun, bis hin zur Vorstufe einer Superintelligenz (wie es z.B. OpenAI-CEO Sam Altman postuliert; hier muss man natürlich auch seine potenziellen wirtschaftlichen Interessen miteinbeziehen).

Was Sprachmodelle aber zweifelsohne von Menschen unterscheidet, ist dass sie nicht in der realen Welt verwurzelt sind. Menschen handeln und sprechen in der echten Welt, ihre Aktionen und Worte haben tatsächliche Auswirkungen auf diese Welt und insbesondere auf ihr eigenes Leben. Diese Komponente fehlt den Sprachmodellen komplett, was als **Grounding Problem** bezeichnet wird.

Bender und Koller (2020) beschreiben dieses Problem anhand eines metaphorischen Oktopus, der aus Gesprächen menschliche Sprachstrukturen lernt. Tagelang belauscht dieser Oktopus zwei Menschen, die auf jeweils unterschiedlichen Inseln gestrandet sind und sich täglich per Telefon unterhalten. Günstigerweise ist das Telefonkabel durch das Unterseewohnzimmer des Oktopus verlegt, sodass er die Kommunikation abhören und irgendwann kapern kann. Er lernt, welche Wörter häufig in welchen Kombinationen vorkommen, und kann tatsächlich das Sprechverhalten des einen Gesprächspartners gut imitieren – allerdings fehlt ihm jeglicher Bezug zu den physischen Objekten auf den Inseln! Selbst wenn man seine „Aussagen“ sinnvoll interpretieren kann, gibt es keine kommunikative Absicht, die damit ausgedrückt werden kann.[5]

Im Kontext dieser Debatte scheint mir das Prinzip von Ockhams Rasiermesser wiederum sehr nützlich – zwischen zwei konkurrierenden Hypothesen ist diejenige zu bevorzugen, die mit weniger (nicht prüfbaren) Annahmen auskommt. Bevor man also einem Sprachmodell menschliche Verhaltensweisen oder tatsächliche Intelligenz andichten will, sollte

[5] Wer wissen will, wie die Geschichte mit dem Oktopus ausgeht, liest am Besten den Abschnitt in dem Artikel selbst – das habe ich aus Platzgründen außen vor gelassen.

man sich die Frage stellen, ob sich die beobachteten Outputs nicht auch einfach aus den technischen Gegebenheiten erklären lassen. Eine nützliche „Checkliste“ hierfür findet sich in (Zweig, 2025, § 31). Ich halte es hier wie Katharina Zweig: Ich habe bislang noch nichts gesehen, das sich nicht aus der technischen Funktionsweise von Sprachmodellen erklären ließe. Die bemerkenswerten Fähigkeiten von heutigen Sprachmodellen sind zwar ein Beleg für die technische Meisterleistung vieler heller Köpfe, aber kein Hinweis auf tatsächliche Intelligenz in Sprachmodellen.

Daher möchte ich auch davor warnen, Sprachmodellen blind zu vertrauen oder sie zu anthropomorphisieren. Sie haben keinen Bezug zur realen Welt, daher kann auch keine **Referenzialität** im wahren Sinne stattfinden. Dementsprechend besteht auch keine kommunikative Intention – diese entsteht erst durch eine „wohlwollende“ menschliche Interpretation. Man denke hier nochmal zurück an den menschlichen Umgang mit ELIZA! Auch wenn Sprachmodelle immer besser werden, sollte man als Nutzer:in stets in der Lage sein, das Output **kritisch einzuordnen** und ggf. zu korrigieren.

Literatur

Bender, E. M., Gebru, T., McMillan-Major, A. & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 acm conference on fairness, accountability, and transparency* (S. 610–623). New York, NY, USA: Association for Computing Machinery. Zugriff auf https://doi.org/10.1145/3442188.3445922 doi: 10.1145/3442188.3445922

Bender, E. M. & Koller, A. (2020, Juli). Climbing towards NLU: On meaning, form, and understanding in the age of data. In D. Jurafsky, J. Chai, N. Schluter & J. Tetreault (Hrsg.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (S. 5185–5198). Online: Association for Computational Linguistics. Zugriff auf https://aclanthology.org/2020.acl-main.463/ doi: 10.18653/v1/2020.acl-main.463

Firth, J. R. (1957). A synopsis of linguistic theory 1930-1955. In *Studies in linguistic analysis*. Philological Society, Oxford. Zugriff auf https://cs.brown.edu/courses/csci2952d/readings/lecture1-firth.pdf (reprinted in Palmer, F. (ed. 1968) Selected Papers of J. R. Firth, Longman, Harlow.)

Harris, Z. S. (1954). Distributional structure. *Word*, *10* (2-3), 146–162. doi: 10.1080/00437956.1954.11659520

Mikolov, T., Yih, W.-t. & Zweig, G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies* (S. 746–751).

Minsky, M. L. (1968). *Semantic information processing*. Cambridge: MIT Press.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Weizenbaum, J. (1966). ELIZA – a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, *9* (1), 36-45. doi: 10.1145/365153.365168

Zweig, K. (2025). *Weiß die KI, dass sie nichts weiß?* München: Heyne.