In den nächsten Jahren werden mehr und mehr KI-Systeme an Schulen und Hochschulen eingesetzt werden. Umso wichtiger ist es, ihren Einsatz vorher gut zu durchdenken und insbesondere auch das sozioinformatische Gesamtgefüge zu verstehen, das durch diesen Einsatz entsteht. In diesem Artikel wird am Beispiel von Sprachmodell-basierten Feedbacksystemen erklärt, warum es dafür zuerst wichtig ist, die grundlegende Technologie zu verstehen: ohne die Beschäftigung kann mit der Technologie nicht bewertet werden, ob KI-basierte Software prinzipiell in der Lage ist, ein Problem zuverlässig zu lösen. Ob der Einsatz von Software insgesamt erfolgreich ist, hängt aber auch von den genauen Bedingungen ab, unter denen sie eingesetzt wird. Daher wird kurz auf die sogenannte sozioinformatische Analyse eingegangen, die versucht, das Verhalten von Personengruppen in Bezug auf ein neu eingeführtes Softwaresystem vorherzusagen.

Schlagworte: Sprachmodell; Feedback; KI in der Schule
Zitiervorschlag: *ZweigKatharina Anna, (2025). Den Einsatz von KI-Systemen mit sozioinformatischen Analysen bewerten. SEMINAR, 31(2), 54-66. Bielefeld: wbv Publikation. https://doi.org/10.3278/SEM2502W005*

E-Journal Einzelbeitrag
von: Katharina Anna Zweig

Den Einsatz von KI-Systemen mit sozioinformatischen Analysen bewerten

aus: Künstliche Intelligenz (KI) im Schulunterricht (SEM2502W)
Erscheinungsjahr: 2025
Seiten: 54 - 66
DOI: 10.3278/SEM2502W005

Den Einsatz von KI-Systemen mit sozioinformatischen Analysen bewerten

Katharina Anna Zweig

Wenn es um KI und Schule geht, wird viel versprochen. Dabei ist es wichtig, für jede einzelne Aufgabe, bei der ein KI-System eingesetzt werden soll, zuerst zu prüfen, ob die zugrundeliegende Technologie überhaupt in der Lage ist, die Aufgabe mit ausreichend hoher Qualität zu lösen. Seit Jahren gibt es beispielsweise den Wunsch, personalisierte Tutorensysteme mit Software umzusetzen. Diese Ideen haben mit dem Fortschritt von „Künstlicher Intelligenz" (KI)-Systemen an Boden.

Kann und sollte ein solches System dazu eingesetzt werden, um beispielsweise Kindern Feedback auf ihre Lösungsansätze zu geben? Viele würden die Frage wie folgend formulieren:

> Kann die KI Feedback auf Schülertexte geben?

Um das zu beurteilen, muss zuerst festgestellt werden, dass es „die KI" gar nicht gibt. Es gibt bisher kein Softwaresystem, dass eine alleinstehende Intelligenz hat, mit der es eigenständig Probleme jeder Art lösen kann. Stattdessen gibt es viele unterschiedliche Softwaresysteme, die verschiedene Aufgaben gut bis sehr gut und teilweise sogar besser als Menschen lösen (vgl. Zweig 2019): Zum Beispiel solche, die benennen, was auf Bildern zu sehen ist (Bilderkennungssysteme); Systeme, die Texte übersetzen (Sprachübersetzer) oder Systeme, die Gesichter oder Fingerabdrücke erkennen und damit Zugang zu Geräten geben. All diese Systeme beruhen auf Methoden der sogenannten *Künstlichen Intelligenz*, aber keins davon ist eine oder gar „die" Künstliche Intelligenz. Die Fragestellung muss daher zuerst präzisiert werden, bevor ergründet werden kann, ob ein bestimmtes KI-System das Problem lösen kann. Dazu scheinen sich die neuartigen Sprachmodelle wie ChatGPT zu eignen. Eine konkretere Formulierung könnte daher sein:

> Kann ein Sprachmodell-basiertes KI-System Feedback auf Schülertexte geben?

Auch diese Formulierung ist noch sehr grob, da nicht spezifiziert wurde, woran man erkennt, ob die Maschine das „kann". Dies könnte man messen am Kompetenzaufbau der Schülerinnen und Schüler für eine bestimmte Aufgabenstellung:

> Kann bei Schülerinnen und Schülern ein Kompetenzaufbau durch Sprachmodell-basierte Feedbacks auf ihre Texte erfolgen?

Dafür muss zuerst die technologische Grundlage recherchiert werden – diese entscheidet darüber, wie hoch die Wahrscheinlichkeit ist, dass ein KI-System eine bestimmte Aufgabe lösen kann. Danach muss der Gesamtkontext des Einsatzes der Software analysiert werden, das sogenannte *sozioinformatische System*. Es besteht meist aus den Nutzerinnen und Nutzer einer Software, sowie weiteren Akteuren (z. B. Schulleitung, Lehrerschaft, Arbeitgeber), der Motivation aller Akteursgruppen und den verschiedenen Eigenschaften, die durch den Einsatz der Software verändert werden. Im Folgenden beschreibe ich daher zuerst die technischen Grundlagen von Sprachmodellen wie ChatGPT, um sie danach darauf zu prüfen, ob sie Feedback auf Texte geben können. Danach behandele ich die Frage, wie sich das sozioinformatische System durch ihren Einsatz verändern könnte.

Technische Grundlagen von Sprachmodellen

ChatGPT beruht auf GPT, einem sogenannten Sprachmodell, das durch das Verarbeiten von riesigen Textmengen gelernt hat, das nächste Wort in einem Textfluss zu erraten. Diese Funktion ahmt das Verhalten des menschlichen Gehirns nach, das ebenfalls ständig versucht vorherzusagen, was passieren wird (Clark 2024). In diesem Sinne werden die Leserinnen und Leser auch vorhersagen, was ich sagen wollte, auch wenn ich mitten im Satz . Hätte ich an dieser Stelle 1.000 Personen gefragt, welches Wort als nächstes wahrscheinlich gewesen wäre, hätten vielleicht 600 Personen „aufhöre“ gesagt, 200 Personen „abbreche“ und weitere 200 „ende“. Damit könnte man für diesen Satzanfang eine Wahrscheinlichkeitstabelle aufstellen: „Menschen würden zu 60 % mit ‚aufhöre‘ ergänzen, zu 20 % mit ‚abbreche‘ und zu 20 % mit ‚ende‘“. Dieses Verfahren der Umfrage und Verwandlung in Prozentzahlen könnte man dann auch noch für jeden beliebigen anderen Satzanfang machen. Es wird aber schnell klar, dass ein solcher Ansatz an zwei Dingen scheitern muss: Am Arbeitsaufwand bei jeder einzelnen Befragung und an der unendlichen Menge an möglichen Satzanfängen, von denen die meisten in der Geschichte der Menschheit noch nie zuvor auch nur gedacht worden sind. Das sogenannte *maschinelle Lernen*, das in riesigen Datenmengen nach generalisierbaren Mustern sucht, ermöglicht aber wenigstens den Versuch dazu.

Das Sprachmodell GPT, die Basis für ChatGPT, basiert auf einem sogenannten *neuronalen Netzwerk*, das auf die folgende Art und Weise trainiert wird: Es bekommt während des Lernens Abschnitte aus extrem umfangreichen Textdokumenten und soll das nächste Wort erraten. Die Antwort kann entweder sehr weit weg vom tatsächlich folgenden Wort liegen oder ziemlich nah dran sein. Die Maschine benötigt hier einen Weg, um diesen „Abstand“ von Antwort und wirklich folgendem Wort zu messen. Das ist nicht trivial, aber wichtig, daher muss ich kurz etwas technischer werden: Ob zwei Wörter „ähnlich“ sind, das ist eine semantische Frage, also eine, bei der man die Bedeutung der Worte kennen müsste. Das tut die Maschine nicht. Stattdessen versucht sie, jedem Wort einen Ort in einem multidimensionalen Raum zuzuweisen, so dass ähnliche Worte nah beieinander sind und unähnliche Worte weit weg voneinander

sind. Diese Zuweisung wird berechnet, indem die Maschine davon ausgeht, dass ähnliche Wörter oft in ähnlichen Kontexten verwendet werden. Ein einfacher Ansatz, der das Prinzip gut erklärt, ist GloVe (Pennington, Socher & Manning 2014). Bei diesem Ansatz wird gezählt, wie oft zwei Wörter in der Nähe von anderen Wörtern in großen Textmengen auftauchen um abzuschätzen, wie ähnlich sie zueinander sind. Wörter wie „Eis“ und „Dampf“ werden in der Nähe von „Wasser“ ähnlich oft auftauchen, weil sie beide mit ihm verwandt sind. Sie werden aber auch ähnlich oft auftauchen in der Nähe des Wortes „Mode“ (nämlich fast nie), weil sie mit diesem Wort wenig zu tun haben. Auf der anderen Seite wird das Wort „Eis“ mit dem Wort „fest“ öfter auftauchen als mit dem Wort „dampfförmig“, weil Eis einen Festzustand bezeichnet. Gesucht wird nun für alle Wörter eine Verortung in einem riesigen, multidimensionalen Raum, so dass Paare von Wörtern, die sich in Bezug auf andere Wörter ähnlich verhalten, auch nah beieinander sind. Da man sich multidimensionale Räume so schlecht vorstellen kann, stelle ich mir hier immer eine Kuppel eines Planetariums vor, aber anstatt Sterne darauf zu projizieren, werden Wörter an die Kuppel projiziert: Jedes Wort bekommt eine Lampe, deren Ausrichtung beliebig geändert werden kann. In dieser Analogie werden die einzelnen Lampen so oft justiert, bis nachher ein Gesamtbild entsteht, so dass die Paare von Wörtern, die ähnlich oft mit dritten Wörtern in einem Text auftauchen, nahe beieinander sind. Das Justieren wird beim maschinellen Lernen automatisiert gemacht: Dabei kann niemand garantieren, dass die Positionierung nachher perfekt ist, aber die Erfahrung zeigt, dass bei langem Training die Ergebnisse nachher oftmals gut genug sind.

Bei den Sprachmodellen kann diese grundsätzliche Positionierung der Wörter entweder vorgegeben oder mitgelernt werden (Raschka 2024). Die Maschine bekommt nun als Input gar nicht die Wörter selbst – sondern nur deren Positionierung in Form einer Reihe von Zahlen. Das ist ähnlich zu den Längen- und Breitengraden, mit denen jeder Ort auf der Welt angegeben werden kann – nur sind es in diesem Fall viel mehr Zahlen (so viele, wie die Anzahl der Dimensionen). Basierend auf diesen Positionierungen soll die Maschine nun raten, was das nächste Wort im Textkontext ist – aber es kann nicht einfach das nehmen, das am nächsten dran ist.

Stattdessen muss sie den Kontext mit „einrechnen“. Dafür gibt es in dem neuronalen Netzwerk viele „Gewichte“. Bei dem Satz „Im Chemielabor explodierte ein Reagenzglas, nachdem in ihm Sauerstoff mit …“ sind die Wörter „Chemielabor, explodierte, Reagenzglas“ und „Sauerstoff“ wichtiger zur Bestimmung des nächsten Wortes als „Im, ein, nachdem“. Genau diese Gewichtung muss die Maschine lernen. Der Mechanismus wird tatsächlich *attention*- oder Aufmerksamkeitsmechanismus genannt – als könnte die Maschine ihre Aufmerksamkeit lenken. Das ist nicht der Fall: Mathematisch gesehen heißt es nur, dass die Maschine lernt, welches Wort im Kontext des Prompts am hilfreichsten dabei sein wird, das nächste mögliche Wort vorherzusagen. Die Positionierungen der Wörter werden anhand der Gewichtungen auf komplizierte Art und Weise miteinander verrechnet – am Ende wird jedem Wort, das die Maschine kennt, eine Zahl zwischen 0 und 1 zugeordnet, die als „Wahrscheinlichkeit“ interpretiert wird.

Aber gerade zu Beginn des Trainings wird sich die Maschine oft vertun, welches Wort im Input wie wichtig ist, um das in Wirklichkeit folgende Wort zu raten. Um eine Verbesserung des Ratens zu erzeugen, muss das System daher während des Trainings kontinuierlich verändert werden: Ein neuronales Netzwerk besteht dabei aus einer Vielzahl von mathematischen Formeln, in denen die jeweiligen Inputs miteinander verrechnet werden – dabei kann einmal der eine Input mehr Gewicht haben, einmal der andere. Diese Gewichtungen können während des Lernens verändert werden. Dabei sollen all diese Gewichte so verändert werden, dass beim nächsten Mal eher das in Wirklichkeit folgende Wort von der Maschine mit einer hohen Wahrscheinlichkeit ausgewählt wird als das jetzige. Durch die ständige Wiederholung dieser Aufgabe und die ständige Anpassung der Gewichtungen wird die Maschine messbar besser, das nächste Wort zu raten. Man hört mit dem Training auf, wenn die Qualität des Ratens ausreichend hoch ist – das liegt natürlich im Auge des Betrachters. Mit dieser Methode lernt die Maschine auf jeden Fall für jeden beliebigen Satzanfang eine „Wahrscheinlichkeit" für das nächste Wort zu berechnen, das in diesem Zusammenhang kommen könnte – und das in einer ziemlich kompakten Form.

Nachdem die Maschine fertig trainiert ist, kann sie mit dieser Methode selbst Texte erstellen: Der Anfangsprompt durch einen Menschen setzt einen Kontext. Dann berechnet die Maschine die Wahrscheinlichkeiten für das erste Wort in der Antwort und wählt dementsprechend eines davon aus. Damit hat es den Kontext um ein Wort verlängert, nimmt den Prompt und das neue Wort als neuen Kontext und rechnet aus, welches das nächstfolgende Wort sein könnte. Auch dieses Wort erweitert den Kontext, der wiederum den Input für das dritte Wort ergibt, und so weiter. Für jedes neu zu generierende Wort wird also der gesamte Anfangsprompt plus die vorher schon generierten Wörter mit eingegeben, um das folgende Wort zu bestimmen. Erfolgt ein weiterer Prompt des Menschen wird auch dieser mit eingebunden. Dies geht so lange, bis die maximale Größe des Prompts erreicht wird, den die Maschine verarbeiten kann – das sogenannte *Kontextfenster*. Passiert dies, werden jeweils nur die letzten Wörter, die gerade noch verarbeitet werden können, verwendet, d. h., die ersten Wörter des Anfangsprompts verschwinden der Reihe nach aus der Eingabe.

Es ist wichtig zu wissen, dass die Maschine nicht immer das Wort mit der größten, berechneten Wahrscheinlichkeit auswählt. Stattdessen gibt es einen Parameter, der je nach Einstellung den vorher errechneten Wahrscheinlichkeiten folgt oder diese eher ignoriert. Diese sogenannte *Temperatur* kann man auf 0 setzen, dann wird immer das „wahrscheinlichste" Wort gewählt. Setzt man ihn auf 2, dann werden beliebige Wörter gewählt. In den meisten Anwendungen sitzt der Parameter auf 1: Damit wird ermöglicht, dass auch als eher unwahrscheinlich markierte Wörter von Zeit zu Zeit verwendet werden, aber alles in allem solche höheren Chancen haben, die auch eine höhere Wahrscheinlichkeit zugesprochen bekommen haben.

GPT ist nun das allgemeine Sprachmodell, das genau diese Funktion erlernt hat. Um es insbesondere für Gespräche im Dialog nutzen zu können, bekam es noch eine weitere Ausbildung: Es wurde mit einem Spezialdatensatz weitertrainiert, in dem Menschen

auf bestimmte Aufforderungen mit einer sinnvollen Antwort reagierten. Danach produzierte es selbst für verschiedene Prompts vier Antworten, die von Menschen in ihrer Sinnhaftigkeit bewertet wurden. Damit wurde eine zweite Maschine trainiert, die daraufhin die Bewertung von Antworten übernahm und damit GPT so veränderte, dass es insbesondere gut im Dialog funktioniert. Mit dieser Methode wurde die Dialogvariante ChatGPT in all seinen Varianten entwickelt. Mehr dazu findet man auf der Webseite von openai: https://openai.com/index/chatgpt/

Die dadurch entstehenden, textgenerierenden Systeme haben auch in der Fachwelt viele Personen überrascht: Denn auch ohne, dass sie das explizit gelernt hätten, können diese Maschine beispielsweise kurze Programme in verschiedenen Programmiersprachen erstellen oder auch Texte sehr gut von einer Sprache in die andere übersetzen. Die Maschine erstellt oftmals sehr gute Texte, aber gerade bei Faktenfragen liegt sie auch oft daneben (s. Zweig 2023). Das verwundert auch nicht, wenn man sich die Funktion der Technologie vor Augen hält: Die Maschine „weiß“ nicht, wann ein Satz genau so und nicht anders generiert werden darf, sondern verwendet ähnliche Wörter in ähnlichen Sätzen. Daher antwortete sie mir auch einmal auf die Frage, wer gerade Kanzler in Deutschland ist, mit „Robert Habeck“. Aber „Kanzler“ und „Vizekanzler“ klingen für die Maschine recht ähnlich, da sie in ähnlichen Kontexten verwendet werden – dementsprechend werden sie eine Positionierung erhalten, die nahe beieinander liegt. Damit kann ein Vizekanzler Habeck, der in Texten rund um Kanzler Scholz häufig auftaucht, auch selbst zum Kanzler werden: Die Maschine weiß nicht, wann welche Wortkombinationen unverrückbar sind und wann in ihnen mathematisch auf diese Weise als ähnlich berechnete Wörter ersetzt werden dürfen.

Sie hat auch keinen Bezug zur Welt: Sie „weiß“ nicht, was eine Brombeere wirklich ist. Das liegt auch daran, dass intern jedes Wort in kleinere Einheiten unterteilt wird, in sogenannte *Tokens*. Das können einzelne Buchstaben oder größere Buchstabengruppen sein, die aber nicht sinntragenden Silben entsprechen müssen. Das Wort „Morgen“ wird von ChatGPT 3.5 zum Beispiel in „M“ und „orgen“ unterteilt, die Brombeere in die vier Token „B-rom-be-ere“. Die Maschine „weiß“ daher noch nicht einmal, dass es sich um ein einziges Ding handelt, sondern hat gelernt, dass auf „B-rom“ in deutschen Texten sehr oft, aber nicht immer, „be“ und „ere“ folgt. Aber natürlich können darauf auch Wörter aus der Welt der Chemie folgen, da das Wort „Brom“ auch ein chemisches Element beschreibt. Die Maschine hat aber kein Metawissen über diese Wörter, da sie nur auf der Ebene von Tokens arbeitet und noch nicht einmal die „kennt“, sondern nur deren Positionierung in einem multidimensionalen Raum. Die Maschine „weiß“ also nicht, dass Brombeeren Früchte sind und Brom meistens ein Gas. Sie kann aber Texte generieren, die Brombeeren als Zutat für einen Obstkuchen vorschlagen, weil Brombeeren im Kontext von „Obst“ viel öfter auftaucht als beispielsweise Brom.

Da aber nun zwei Wörter ähnlich sind, wenn sie in ähnlichen Kontexten auftauchen, kann damit ein Vizekanzler Habeck, der in Texten rund um Kanzler Scholz häufig auftaucht, auch selbst zum Kanzler werden.

Daher wird eine auf diese Art entwickelte Maschine immer *konfabulieren*, also Texte generieren, die nicht der Wahrheit entsprechen – denn rein technisch gesehen hat sie kein Wissen darüber, was sie eigentlich tut. Und entgegen vieler Formulierungen, die einem rund um ChatGPT und andere Sprachmodelle begegnen, hat sie auch keine „Wissensdatenbank", sondern einfach nur eine riesenhafte Menge an Texten verarbeitet, um zu lernen, was das wahrscheinlichste nächste Wort (bzw. Token) in einem bestimmten Kontext ist.

Nicht zuletzt lernt die Maschine durch diese Art des Trainings natürlich auch Stereotypen: Wenn ein Sachverhalt immer und immer wieder auf ähnliche Weise dargestellt wird, obwohl dies nur ein Vorurteil ist oder gar gänzlich falsch, dann wird die Maschine diesen Sachverhalt auch ähnlich darstellen. Sie kann das von ihr generierte in keiner Form bewerten oder bewusst formen. Daher sprechen manche Wissenschaftlerinnen und Wissenschaftler von Sprachmodellen als „stochastische Papageien" („stochastic parrots"):

> „Ein Sprachmodell (LM) ist ein System, das auf zufällige Weise sprachliche Formen, die es in seinen umfangreichen Trainingsdaten beobachtet hat, zu Sequenzen zusammensetzt. Dabei nutzt es statistische Informationen darüber, wie diese sich dort kombinieren ließen, jedoch ohne jeglichen Bezug zur Bedeutung: ein stochastischer Papagei." (Bender et al. 2021)

Zusammenfassung technische Grundlagen: Ein Sprachmodell ergänzt Sätze nach den intern berechneten Wahrscheinlichkeiten, wobei die berechneten Wahrscheinlichkeiten während des Trainings so justiert wurden, dass es oftmals das tatsächlich folgende Wort in einem Text gut erraten konnte. Eine höhere Einstellung der „Temperatur" ermöglicht, dass nicht immer nur das am wahrscheinlichsten berechnete Wort als nächstes genommen wird. Die Maschine hat durch ihr extensives Training keine Wissensdatenbank zur Verfügung, aber wenn in vielen Texten dieselben oder sehr ähnliche Wörter immer wieder zur Beschreibung eines Sachverhaltes verwendet werden, wird sie diesen Sachverhalt vermutlich in ähnlicher Weise wiedergeben.

Wenn die technischen Grundlagen soweit bekannt sind, kann die eigentliche Analyse erfolgen, ob ein Softwaresystem für eine spezifische Frage eingesetzt werden kann. Diese Analyse erfolgt im nächsten Abschnitt für die Frage, ob Sprachmodell-basierte KI-Systeme Feedback auf Schülertexte geben können.

Können Sprachmodell-basierte KI-Systeme Feedback auf Schülertexte geben?

Ein Feedback stellt eine Entscheidung über einen Text dar. Ich unterscheide bei sogenannten *automatisierten Entscheidungs-(unterstützungs-)Systemen* zwischen vier verschiedenen Sorten von Entscheidungen: Faktische Entscheidungen, Vorhersagen, singuläre Entscheidungen und Werturteile (*judgments*). Ein Feedback gehört dabei zu den

Werturteilen, die nach Kahnemann, Sibony und Sunstein (2022) so definiert sind, dass sich Experten und Expertinnen bei ihrem Urteil nicht beliebig uneinig sein dürfen. Müssten sie sich einig sein, wäre es ein Fakt; müssten sie sich gar nicht einig sein, wäre es eine Meinung. Ein Werturteil charakterisiert sich dadurch, dass 1) Maßstäbe der Beurteilung festgelegt werden müssen, 2) das Ausmaß der Erfüllung dieser Maßstäbe bewertet werden muss und optional 3) daraus eine Gesamtbewertung durch Gewichtung der einzelnen Qualitätsdimensionen und ihrer Erfüllung hergestellt wird. Bei einem Feedback muss dann natürlich noch viel Wert auf die Kommunikation des Ergebnisses gelegt werden und dabei beispielsweise Lernstand und Alter einer Schülerin oder eines Schülers berücksichtig werden.

Ein Sprachmodell kann ohne Frage Texte erstellen, die der Struktur eines Feedbacks folgen. Man kann die Maschine durch geeignete Prompts auch dazu bringen, diese Texte für verschiedene Leserschaften aufzubereiten: Beides sind Aufgaben, die rein struktureller Natur sind. Die Maschine hat in den großen Datenmengen anscheinend ausreichend viele Beurteilungen und Rückmeldungen gesehen, um den Duktus dieser Texte im wahrsten Sinne zu verinnerlichen, d. h., in passende Gewichtungen zu verwandeln. Sie kann aber Qualitätsmaßstäbe nicht selbst identifizieren. Das Nennen von Qualitätsmaßstäben im Prompt kann daher nur Assoziationen wecken: „Gib den Schülern ein Feedback auf ihre Texte. Berücksichtige dabei Rechtschreibung, Kohärenz und Vielseitigkeit der Argumente.“. Ein solcher Prompt wird dazu führen, dass die Maschine in ihrem „Feedback“ die Rechtschreibung, Kohärenz und Vielseitigkeit der Argumente explizit erwähnen wird. Warum? Weil sie gelernt hat, dass solche Aufforderungen Antworten benötigen, in denen die genannten Begriffe wiederholt werden – das hat sie in ihrem „Aufmerksamkeitsmechanismus“ gespeichert. Sie wird in den meisten Fällen auch einen in sich konsistenten Text schreiben, d. h., wenn sie lobend die Rechtschreibung und die Kohärenz erwähnt, dann wird vermutlich auch die Vielseitigkeit der Argumente lobende Worte finden. Das liegt daran, dass sehr oft Bewertungen entweder an allen Stellen gut sind oder an allen Stellen weniger gut. Die Maschine „weiß“ aber nicht, ob ein zu beurteilender Text kohärent geschrieben ist. Die Maschine „weiß“ noch nicht einmal, ob die Rechtschreibung korrekt ist oder nicht. Das erzeugt oft Verwunderung, da die Maschine ja sicherlich viel öfter das Wort „Gefahr“ als das Wort „Gefar“ gelesen hat und daher doch sicher „wissen“ muss, dass „Gefahr“ die richtige Schreibweise ist. Es ist korrekt, dass die Maschine selbst das Wort „Gefar“ nicht schreiben würde, weil diese Kombination von Tokens eine kleinere Wahrscheinlichkeit hat als „Gefahr“. Aber die Maschine „weiß“ nicht, dass dies der Grund ist, warum sie „Gefahr“ statt „Gefar“ schreibt. Wird sie gefragt, warum Gefahr die richtige Schreibweise ist, denkt sie sich a) etwas aus und b) behauptet sie das einfach. Ich habe sie gefragt und das war ihre Antwort:

> „Ich weiß das, weil ‚Gefahr‘ ein deutsches Wort ist, das korrekt mit h geschrieben wird. Es stammt vom althochdeutschen Wort *gifara*, was „das, was fährt“ bedeutete. Das h ist historisch bedingt und gehört zur standardmäßigen Rechtschreibung im Deutschen.
>
> „‚Gefar‘ gibt es in der deutschen Sprache nicht – es wäre ein Rechtschreibfehler. 😉 “

Die eigentliche Begründung („Gelernte Wahrscheinlichkeiten aus großem Datensatz mit Methoden der künstlichen Intelligenz“) kennt die Maschine nicht – sie hat kein Metawissen über sich selbst. Die angegebene Ableitung von „Gefahr“ aus *gifara* ist falsch und ansonsten behauptet das System einfach zweimal, dass es so sei: „ich weiß das, weil das korrekt mit h geschrieben wird“ und „Gefar wäre ein Rechtschreibfehler“. Aussagen ohne Argument.

Die Maschine „weiß“ auch nicht, ob sie gerade etwas lobt oder nicht. Sie hat nur Assoziationen zwischen bestimmten Wendungen und Wortzusammenstellungen gelernt. Dabei ist es durchaus möglich, dass die Maschine Korrelationen zwischen Texten mit (aus menschlicher Sicht) „lobendem Feedback“ und „kritischem Feedback“ findet. So werden Texte im Training mit Wörtern in häufiger Schreibweise vermutlich oft als Rückmeldung bekommen „Gute Rechtschreibung“ als solche Texte mit Wörtern in seltener (falscher) Schreibweise. Ein Text, der viele Fremdwörter enthält, wird vielleicht auch öfter ein „lobendes Feedback“ bekommen als einer, der nur sehr kurze Hauptsätze enthält. Beides sind strukturelle Elemente, die die Maschine durch das häufige, gemeinsame Auftauchen durchaus lernen kann (vgl. Schneider & Zweig 2023).

Schneider nennt das Erstellen solcher Texte, die beim Lesen von Menschen mit Interpretation und menschlicher Intelligenz angereichert werden als „intelligible Texturen“ (Schneider 2024): „Um dies zu verdeutlichen, gebrauche ich hier den von Christian Stetter (1997, 295 ff.) eingeführten Begriff ‚Textur‘: Texturen sind materielle Zeichengebilde, also z. B. Buchstaben auf Papier, die erst dadurch, dass sie gelesen und verstanden werden, zu Texten werden. Insofern lässt sich sagen, dass ChatGPT intelligible Texturen erzeugt – Gebilde, die sich als intelligente Texte lesen und interpretieren lassen.“

Die Frage, ob Sprachmodell-basierte KI-Systeme Feedback auf Schülertexte geben können, reduziert sich damit auf die Frage, ob ein Schülertext im Allgemeinen ausreichend Assoziationen bei der Maschine weckt, um mit hoher Wahrscheinlichkeit ein passendes Feedback zu erzeugen. Gerade im Grundschulbereich ist dies denkbar, da es hier Feedbackteile gibt, die vermutlich fast jeder Text bekommt: „Achte auf Rechtschreibung und Groß- und Kleinschreibung“ oder „Fang nicht jeden Satz mit ‚dann‘ an und ergänze mehr Nebensätze“, beispielsweise, weil diese Fähigkeiten oft noch nicht sehr stark ausgeprägt sind. Die Qualität des Feedbacks müsste damit also untersucht werden, und zwar am besten in direktem Vergleich mit der Qualität des Feedbacks von geschultem Lehrpersonal.

Ob und inwieweit es solche Studien für welche Art von KI-System heute schon gibt, ist nicht relevant für den vorliegenden Artikel: Wichtig ist, dass die Studien zur eigenen Bildungslandschaft passen, z. B. in Deutschland und bestenfalls in demselben Bundesland durchgeführt wurden und dass sie zudem auch eine mindestens ähnliche Fragestellung betrachten, z. B. Feedback auf Gedichtanalysen in Klassen 7–10 oder Feedback auf Mathehausaufgaben für schriftliche Multiplikation und Division. Wichtig ist auch, dass die Qualität des Feedbacks von geschulten Experten und Expertinnen bewertet wurde, und nicht nur die Schüler und Schülerinnen gefragt wurden, ob es ihnen gehol-

fen hat. Am hilfreichsten wäre eine Einschätzung, wie oft ein Feedback wie stark zum zu bewertenden Text passt – am besten noch im Vergleich zu menschlich erstelltem Feedback, das auf dieselbe Art und Weise von den Experten und Expertinnen bewertet wird.

Erst mit einem solchen Vergleich von menschlichen und maschinell erstellten Feedbacks könnte man in die nächste Phase der Analyse starten, nämlich in die Frage, wie die Nutzung des Systems im Unterricht oder zu Hause gestaltet werden sollte, also wie das *sozioinformatische System* aussieht.

Analyse des sozioinformatischen Systems bei der Nutzung von Sprachmodell-basierten KI-Systemen zur Feedbackgenerierung

Wann immer Menschen Technik verwenden, entsteht durch diese Verwendung ein soziotechnisches System: Dabei erlaubt die Technologie manche Prozesse schneller oder einfacher durchzuführen, kann dabei aber auch andere Prozesse erschweren und insgesamt Eigenschaften des Gesamtsystems verändern. So erlaubt uns das Auto einen relativ günstigen Individualverkehr, nimmt in Städten aber durch die Angewiesenheit auf Straßen auch relativ viel Platz ein. Wenn nun die Technologie eine informatische Technologie ist, nennen wir die entstehenden Systeme *sozioinformatische Systeme* (Zweig et al., 2021). Die Modellierung und Analyse solcher Systeme zu lehren ist der Zweck des Studiengangs *Sozioinformatik* an der RPTU, der in dieser Form in Deutschland bislang einmalig ist.

Sozioinformatische Systeme bestehen aus Gruppen von Menschen, die wir als Akteure bezeichnen und die eine Motivation haben, welche von den Veränderungen durch den Einsatz einer Software betroffen ist. Dabei ist es schwierig, eine genaue Grenze zu ziehen, wer genau vom Einsatz einer Software betroffen ist: Bei einigen Softwaresystemen wie den verschiedenen Social-Media-Plattformen, stellte sich über die Zeit heraus, dass die Gesellschaft als Ganzes betroffen war und daher die nationalen und europäischen Gesetzgeber einschritten. Diese Akteure waren somit betroffen, aber nur, wenn man den langen Zeitraum betrachtet. Wir sprechen daher vom *Zeithorizont* einer sozioinformatischen Analyse, der grob festlegt, wie lang der Zeitraum sein soll, für den die Analyse gilt. Für das in diesem Artikel zugrundeliegende Problem könnte die Frage präzisiert werden: „Welchen Einfluss auf die Kompetenzbildung hat die Verwendung von Sprachmodell-basierten System zur Generierung von Feedback an Gymnasien?“. Dies grenzt den Zeithorizont die Zeit ein, die normalerweise benötigt wird, um eine bestimmte Kompetenz auszubilden. Die Menge der betroffenen Akteure könnte groß sein: z. B. Lehrpersonen, Schulleitung, Schülerschaft und die nachfolgenden Arbeitgeber und/oder Ausbildungsstätten. Hier geht es mir aber primär um den Effekt auf die Lehrerinnen und Lehrer und damit wähle ich nur diese Gruppe aus.

Die kurz umrissene Diskussion zur Auswahl der Akteure zeigt, dass das Erstellen eines sozioinformatischen Systems eher eine Modellierung ist als eine Methode, die zu einem einzigartigen Resultat führt (Zweig et al. 2021). Bei einer Modellierung abstrahiert man von einem komplexen System all das, was für einen vorher definierten Zweck dienlich sein könnte. Das Modell, das in einer sozioinformatischen Analyse im Mittelpunkt steht, dient dem Zweck, die Hauptdynamiken im sozioinformatischen System zu verstehen. Was genau die Hauptdynamiken sind, hängt dabei immer auch vom menschlichen Werturteil ab, das unterschiedliche Foki setzen kann. Hat man sich auf diese geeinigt, wird im nächsten Schritt ein sogenanntes Wirkungsgefüge erstellt, dass für die Akteure und eine Reihe von Parametern darstellt, wie diese aufeinander einwirken. Wir gehen dabei davon aus, dass jede Akteursgruppe Motivationen mit sich bringt: Diese stellen wir dar als Systemparameter, deren Wert eine Gruppe erhöhen oder erniedrigen will.

Daher identifiziert man als Nächstes für die betrachteten Akteursgruppe(n) deren jeweilige Hauptmotivationen. Hier habe ich vereinfachend zwei Motivationen der Lehrer gewählt: „Kompetenzausbildung der SuS“, dazu aber auch den Systemparameter „verfügbare Zeit“: Ich gehe davon aus, dass Lehrer und Lehrerinnen davon gerne mehr hätten. Es ist nun bekannt, dass persönliches Feedback für eine Kompetenzausbildung hilfreich wäre – aber dieses persönliche Feedback zu geben verkürzt gleichzeitig die verfügbare Zeit für andere Tätigkeiten. Es gibt nun also schon drei Systemparameter, die wir in einem Wirkungsgefüge (siehe Abbildung 1) repräsentieren können: Motivationen werden dabei durch Kanten zwischen einer Akteursgruppe und des zu erhöhenden oder zu erniedrigenden Parameters als Kante gezeichnet. Um sie als Motivation zu kennzeichnen, bleibt es bei einem Strich. Parameter, die aufeinander einwirken, werden mit einem Pfeil miteinander verbunden. Der Pfeil drückt aus: „Wenn der erste Parameter sich verändert, ändert das den zweiten Parameter“. In dieser sehr einfachen Modellierung unterscheiden wir nur zwischen zwei Interaktionen: „Je mehr von A, desto mehr von B“ (oder: „Je weniger von A, desto weniger von B“) – wir sprechen von einer agonistischen Wechselwirkung. Die zweite drückt aus: „Je mehr von A, desto weniger von B“ (oder: „Je weniger von A desto mehr von B“) – dies ist eine antagonistische Wechselwirkung. Um diese Arten der Interaktion voneinander unterscheiden zu können, ist der eine Pfeil durchgezogen, der andere ist gestrichelt.

Jetzt ist natürlich die Frage, ob Sprachmodell-basierte Modelle mit dem Erstellen der Feedbacks helfen können: Damit können wir nun noch eine weitere Systemvariable einführen, nämlich unten links die „Anzahl personalisierter Feedbacks durch KI“. Aus den technischen Grundlagen ist nun bekannt, dass die Antworten der KI nicht immer hilfreich und korrekt sind. In einem guten Bildungsprozess sollte dies auch den Kindern bekannt gemacht werden mit der Ansage, dass diese bei vermeintlich falschen Feedbacks zur Lehrkraft gehen sollen, um dies gemeinsam zu besprechen. Die Anzahl der vermeintlich falschen Feedbacks erniedrigt aber wieder die zur Verfügung stehende Zeit für die Lehrkräfte – unabhängig davon, ob sie wirklich falsch sind oder nur fälschlich als nicht hilfreich empfunden wurden. Natürlich kann man auch erwarten,

dass der Einsatz von KI zu mehr hilfreichen persönlichen Feedbacks führt, die dann ebenfalls dem Kompetenzaufbau dienen. Es gibt aber auch eine dritte Gruppe: irreführende Feedbacks, die von den Kindern nicht erkannt werden, und dann vielleicht sogar zu Kompetenzverlust führen.

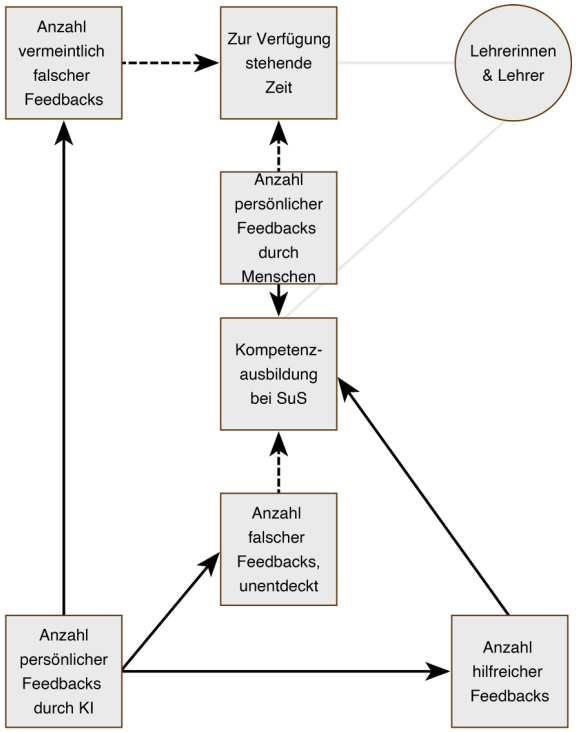

Abbildung 1: Sozioinformatisches Wirkungsgefüge für die Nutzung von Sprachmodell-basierten Feedbacksystemen. Es gibt die Wechselwirkungen zwischen bestimmten Systemparametern (Quadrate) und der Motivation der Lehrkräfte wieder.

Das Wirkungsgefüge hilft nun dabei, die Hauptdynamiken bei der Nutzung von sprachmodell-basierten Feedbacksystemen zu identifizieren. Aus den technischen Grundlagen ist nur bekannt, dass die Maschine auch unpassende Feedbacks geben wird. Nicht bekannt ist, wie gut Schülerinnen und Schüler darin sein werden, hilfreiche und nicht hilfreiche Feedbacks zu unterscheiden. Davon hängt aber ab, ob Kompetenzaufbau mit ihnen gelingen kann und wieviel Zeit der Lehrkraft bleibt: denn wirklich falsche Feedbacks benötigen Korrektur, wenn solche, die nicht korrigiert werden, einen Schaden hinterlassen.

Die Frage, ob bei Schülerinnen und Schülern ein Kompetenzaufbau durch Sprachmodell-basierte Feedbacks auf ihre Texte erfolgen kann, wird nun durch das Wirkungsgefüge nicht abschließend geklärt, aber es wird sichtbar, dass zur Beantwortung der Frage das Gesamtsystem mit seinen Einschränkungen betrachtet werden muss. Mehr

Feedback ist nicht unbedingt gleichbedeutend mit mehr Kompetenzaufbau – der Zusammenhang ist nicht linear. Stattdessen wird der Effekt moduliert von anderen Aspekten, nämlich dem Vermögen der Schülerinnen und Schüler zwischen hilfreichen und nicht hilfreichen Feedbacks zu unterscheiden; dies wirkt wiederum auf den Zeitdruck der Lehrkräfte. Die Visualisierung dieser Zusammenhänge hilft dabei, Diskussionen um den Einsatz von Software immer wieder auf die wesentlichen Zusammenhänge zu lenken. Mit Hilfe des Wirkungsgefüges wird dann auch klar, was momentan unbekannt ist und gemessen werden sollte, wenn die Software zum Einsatz kommt: Nämlich der Kompetenzaufbau in Abhängigkeit davon, wieviele Feedbacks mit der Lehrkraft besprochen werden müssen in Abhängigkeit davon, wie gut die Kinder hilfreiche und nicht hilfreiche Feedbacks unterscheiden können.

Zusammenfassung sozioinformatische Analysen in der Bildung

Eine sozioinformatische Analyse versucht zu illustrieren, wie sich ein soziales System aus Akteuren und Systemparametern durch den Einsatz von Software verändert. Diese Analyse besteht immer aus den folgenden Schritten (Zweig et al. 2021): Zuerst muss die Fragestellung eruiert werden. Danach muss die Technologie betrachtet werden: Welche Hinweise darauf, dass die Technologie ein Problem lösen kann, gibt es? Welche weiteren Tätigkeiten könnte sie erleichtern oder erschweren, die dann wiederum Auswirkungen auf Systemparameter haben könnten? Als dritter Schritt erfolgt die Auswahl der hauptsächlich betroffenen Gruppen, der Akteure und deren Motivationen: Welche Systemparameter wollen diese erhöht sehen, welche erniedrigt? Im vierten Schritt werden alle Akteure, deren Motivationen und weitere Systemparameter und eine sehr einfache Form von Wechselwirkung (agonistisch/antagonistisch) in ein visuell darstellbares Wirkungsgefüge gebracht. Dieses kann im fünften Schritt analysiert werden: Können alle Akteure ihre eigenen Motivationen mit Hilfe der Software durchsetzen oder werden sie darin gebremst? Können sie „Nebenwege“ nutzen, um ihren Motivationen zu folgen?

Im Mittelpunkt der Analysen stehen dabei insbesondere die indirekten Auswirkungen, weil die einfache Frage, ob Software X Problem Y lösen kann, einfach nicht ausreicht, wenn Menschen in verschiedenen sozialen Prozessen miteinander und mit der Software interagieren. Das System Schule mit seiner rechtlichen Basis, den schuleigenen Regeln und nicht zuletzt den verschiedenen Bedürfnissen der Individuen ist eines, in dem die neuen, KI-basierten Softwaresysteme viel positiv verändern könnten – aber nur, wenn die sozialen Prozesse mit in den Blick genommen und bei Bedarf angepasst werden. Die sozioinformatische Analyse ermöglicht diesen erweiterten Blick auf den Einsatz von Software.

Literaturverzeichnis

Bender, E. M.; Gebru, T; McMillan-Major, A.; Shmitchell, S. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Im Konferenzband der 2021 ACM Conference on Fairness, Accountability and Transparency (FAccT '21). 610–623.

Clark, A. (2024). The experience machine – how our minds predict and shape reality. UK: Penguin Random House.

Kahnemann, D; Sibony, O.; Sunstein, C. R. (2021). Noise: Was unsere Entscheidungen verzerrt – und wie wir sie verbessern können. München: Siedler Verlag.

Pennington, J.; Socher, R.; Manning, C. (2014). GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar: Association for Computational Linguistics. 1532–1543.

Schneider, J. G.; Zweig, K. A. (2023). Grade Prediction is not Grading. In Groß, R.; Jordan, R. (Hrsg.). „KI-Realitäten". transcript Verlag. 93–11. Öffentlich verfügbar unter https://www.degruyter.com/document/doi/10.1515/9783839466605–005/html

Schneider, J. G. (2024). Intelligible Texturen. Welche Rolle kann ChatGPT bei der Aufsatzbewertung spielen?. VK:KIWA. https://doi.org/10.5281/zenodo.10877034

Zweig, K. A. (2019). „Ein Algorithmus hat kein Taktgefühl", München: Heyne-Verlag.

Zweig, K. A.; Krafft, T.; Klingel, A.; Park, E. (2021). Sozioinformatik – ein neuer Blick auf Informatik und Gesellschaft. München: Hanser Verlag.

Zweig, K. A. (2023). Die KI war's. München: Heyne-Verlag.

Prof. Dr. Katharina Anna Zweig
ist Professorin an der RPTU in Kaiserslautern, wo sie das Algorithm Accountability Lab und den Studiengang Sozioinformatik leitet.