BENUTZERMODELL ZUR SIMULATION VON INTERAKTIONEN MIT SPRACHDIALOGSYSTEMEN BASIEREND AUF AKTIVIERUNG VON TEIL-ZIELEN

Klaus-Peter Engelbrecht

Quality and Usability Lab, Telekom Innovation Laboratories, TU Berlin
Klaus-Peter.Engelbrecht@telekom.de

Abstract: Benutzermodelle werden heute bei der Evaluierung von Sprachdialogsystemen, sowie beim automatischen Lernen solcher Systeme aus Daten genutzt, um mögliche Dialogverläufe zu simulieren. Bisher wurden zumeist Modelle genutzt, die das Benutzerverhalten statistisch beschreiben. Um jedoch auch Informationen über das subjektive Empfinden der Benutzer während der Interaktion schätzen zu können, ist die Modellierung kognitiver Prozesse sinnvoll. In diesem Paper wird daher ein neuer Modellierungsansatz basierend auf der Aktivierung von Teilzielen auf Basis der vorangehenden Systemaussage und des Kontextes vorgestellt. Das Modell wird mit einem State-of-the-Art-Ansatz zur Benutzermodellierung verglichen.

1 Einleitung

Sprachdialogsysteme (SDS) sollen eine natürliche und effiziente Interaktion zwischen Menschen und Computern ermöglichen. Leider kommt es bei der Interaktion jedoch häufig zu Fehlern beim Verstehen oder bei der Interpretation der Benutzeräußerungen, sowie zu unvorhergesehenen Situationen im Dialogverlauf. Das Design solcher Systeme ist daher komplex und erfordert umfassende Tests des Systems.

In der Forschung werden bereits vielfach Benutzermodelle zur simulationsbasierten Evaluierung von Sprachdialogsystemen eingesetzt [1][2][3]. Andere Forscher versuchen, Dialogstrategien automatisch zu lernen, greifen dabei jedoch auch auf Benutzermodelle zurück, um genügend Trainings-Material erzeugen zu können (z. B. [4][5]). Eine Reihe von Modellen ist in diesem Kontext entstanden, bei denen Benutzerverhalten zumeist statistisch beschrieben wird.

Während sich die Modelle bei der formativen Evaluierung bereits bewährt haben [1][2][6], lassen sich Interaktionsparameter wie die durchschnittliche Ausführungszeit oder der durchschnittliche Aufgabenerfolg nur mäßig mit solchen Modellen prognostizieren [6]. Zudem wurden bisher keine Modelle gefunden, die sich direkt für beliebige (oder eine größere Klasse von) Systeme(n) einsetzen lassen. Hier besteht also direkter Bedarf an valideren und zumindest teilweise kausal arbeitenden Modellen.

Schließlich ist ein prinzipieller Nachteil der Evaluation mit Benutzermodellen, dass keine subjektiven Größen wie Zufriedenheit gemessen werden können. Hierfür bestehen jedoch auch Modellierungsansätze, insbesondere PARADISE [7]. Leider ist auch hier bisher die Vorhersagegenauigkeit für praktische Anwendungen, bei denen die Modelle auf neue Systeme angewandt werden, nicht ausreichend [6]. Beide Forschungsgebiete könnten davon profitieren, bestimmte Aspekte von Kognition, wie z. B. Motivationen und Affekte, explizit zu modellieren. Da sich Beurteilung und Verhalten auch beeinflussen können [8], liegt zudem eine Integration der Modelle nahe.

Obwohl für solche integrierten Modelle eine Simulation auf kognitiver Ebene naheliegt, sind kognitive Architekturen wie ACT-R [9] für diese Anwendung nur bedingt geeignet, da hier zunächst nur die Simulation von rationalem Verhalten vorgesehen ist. Komplexere

Phänomene wie Motivation oder Emotionen könnten möglicherweise auf Basis von Wissenseinheiten („Chunks“) und Produktionsregeln definiert werden, jedoch wäre dies sehr aufwändig, und die Architektur stellt keinerlei Wissen oder Mechanismen bereit, die dies unterstützen könnten. Die PSI-Theorie [10] stellt solches Wissen bereit, allerdings wird hier die Entwicklung eines Agenten auf einer Insel simuliert, die v.a. durch grundlegende Lebenserhaltungsprozesse (Hunger, Durst, Schmerz) geprägt ist. Für den Kontext Mensch-Maschine-Interaktion sind daher viele Annahmen nicht sinnvoll (z. B. der Einfluss von Hunger auf die Ziel-Auswahl). Jedoch sind hier auch grundlegende Algorithmen für Motivation und Problemlösen spezifiziert, so dass die PSI-Theorie potentiell eine geeignete Grundlage für die Modellierung eines affektiv sich verhaltenden Benutzers bereitstellt.

Nach der PSI-Theorie werden Ziele durch Bedürfnisse wie Hunger, Durst oder Affiliation aktiviert. So könnte das Ziel „Wasser trinken“ durch das Bedürfnis Durst aktiviert werden. In der Regel sind mehrere konkurrierende Ziele gleichzeitig, aber in unterschiedlichem Maße aktiviert. Für die Auswahl des zu verfolgenden Ziels ist zunächst von Bedeutung, wie stark es aktiviert ist, aber auch, wie leicht es zu erreichen ist. Zudem gibt es eine Selektionsschwelle, die das Wechseln von Zielen reguliert.

Auch neuere Arbeit zur Organisation von Teilzielen in ACT-R gehen davon aus, dass diese – ähnlich wie der Abruf deklarativen Wissens aus dem Gedächtnis – durch Aktivierungsprozesse geregelt wird [11]. Die Aktivierung von Teilzielen setzt sich demnach, genau wie die Aktivierung von Chunks, aus einer Grundaktivierung und einer kontextabhängigen Aktivierung zusammen. Ist ein Teilziel erreicht, wird als nächstes das Ziel mit der höchsten Gesamtaktivierung fokussiert.

Dieses Prinzip findet sich im Zusammenhang mit Sprachdialogen auch bei Putze und Schultz [12]. Zunächst wird hier aufgegriffen, dass bestimmte Gedächtnisinhalte (bzw. Chunks) durch Umgebungsreize aktiviert werden und anschließend als Gesprächsinhalte relevant werden. Die Aktivierung kann zudem durch ein assoziatives Netz propagiert werden und verfällt mit der Zeit. Putze verwendet dieses Prinzip zur Interaktionsauswahl bei einem Sprachbasierten Tour-Guide, jedoch wird hier keine Zielführende Interaktion angestrebt, sondern die Unterhaltung des Klienten.

Von diesen Ansätzen inspiriert soll in diesem Paper untersucht werden, inwieweit beobachtbares Benutzerverhalten auf Konzeptebene durch das Konzept der Aktivierung und Selektion von Teil-Zielen erklärt werden kann. Dazu wird zunächst das Agenda-basierte Benutzermodell [4], das zur Zeit den State-of-the-Art bei der Simulation von Dialogen mit Gemischte-Initiative-Systemen darstellt, in ein äquivalentes, aber auf Aktivierung basierendes Modell übersetzt. Damit soll gezeigt werden, dass ein auf Aktivierung basierendes Model die selbe Ausdrucksstärke hat wie das auf der Agenda basierende. Der Ansatz wird daraufhin weiter ausgearbeitet und evaluiert Nächste Schritte im Hinblick auf die Simulation von Emotionen werden diskutiert.

2 Statistische Modellierung von Benutzerverhalten und das Agenda-basierte Benutzermodell

2.1 Grundlagen der statistischen Benutzermodellierung

Wie in Abbildung 1 zu sehen, kann die Simulation von Dialogen mit einem SDS auf unterschiedlichen Ebenen erfolgen. Insbesondere sind Benutzereingaben auf der Ebene von Dialogakten, Text, oder gesprochener Sprache möglich. Die Auswahl der Simulationsebene hängt dabei davon ab, welche Komponenten auf der Eingabeseite des Systems in die Simulation einbezogen werden sollen. Sol z. B. die automatische Spracherkennung (ASR) mitgetestet werden, ist eine Simulation auf Sprach-Ebene erforderlich. Bei Simulationen auf

einer tieferen Ebene werden Fehler bei der ASR, sowie beim Sprachverstehen (NLU) durch entsprechende Modelle simuliert, da diese sich auf die darunterliegenden Komponenten auswirken können. Wie in der Abbildung zu sehen, erfolgt die Kommunikation vom System zum Benutzer in der Regel auf der Ebene von Sprechakten, da von einem fast perfekten Verstehen auf der Benutzerseite ausgegangen werden kann.

Bei der Spezifikation von Benutzermodellen steht in der Regel die Auswahl von Dialogakten im Fokus, während die Sprachgenerierung, genau wie beim SDS, entkoppelt als nachfolgendes Modul aufgefasst wird. Die Modellierungsaufgabe besteht bei einer statistischen Modellierung also darin, die Wahrscheinlichkeitsverteilung der möglichen Benutzeraktionen in einer bestimmten Dialogsituation zu spezifizieren. Dabei lässt sich die Situation im einfachsten Fall durch die vorangehende Systemausgabe (a_{sys}), sowie die gestellte Aufgabe (T) beschreiben. Um konsistenteres Benutzerverhalten zu erzeugen, wird jedoch in der Regel der bisherige Dialogverlauf in die Situationsbeschreibung aufgenommen. Die Kunst besteht nun darin, den bisherigen Dialogverlauf möglichst kompakt (d.h. mit wenigen Parametern) zu beschreiben, um diese leicht spezifizieren oder aus wenigen Daten berechnen zu können.

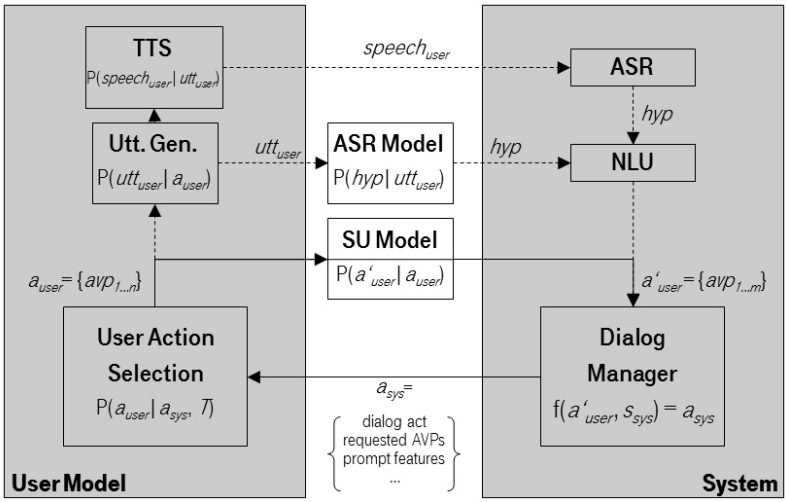

Abbildung 1 – Prinzipieller Aufbau bei der Simulation von Dialogen mit statistischen Benutzermodellen.

2.2 Das Agenda-basierte Benutzermodell

Beim Agenda-basierten Benutzermodell (AGM) werden die relevanten Aspekte des bisherigen Dialogverlaufs in der „Agenda“ gespeichert. Die Agenda ist eine Datenstruktur vom Typ Stack (Stapel), die die Reihenfolge der Ausführung von Teilzielen zur Erledigung einer übergeordneten Aufgabe organisiert. Gegeben eine Situation, werden Teilziele in den Stack „gepusht“, d.h., obenauf gelegt. Um eine Benutzeraktion zu erzeugen, wird eine bestimmte Anzahl n von Teilzielen „gepoppt“, d.h., von oben abgenommen. Zuletzt aufgetretene Teilziele haben also immer Priorität vor früher aufgetretenen Teilzielen, sofern letztere nicht erneut in den Stack gepusht werden (in diesem Fall wird die tieferliegende Kopie gelöscht). Teilziele sind dabei einzelne Sprechakte, oder im hier beschriebenen Fall Attribut-Wert-Paare (AVPs), d.h. einzelne semantische Konzepte, die die Intention des Benutzers beschreiben. Sprechakte sind in den meisten Fällen äquivalent zu Attribut-Wert-Paaren, so dass dieser Unterschied vernachlässigbar ist. Teilweise werden diese elementaren Benutzeraktionen auch direkt als „Ziele“ bezeichnet (z. B. [2]). Das „poppen“ und „pushen“ von Teilzielen während eines Dialogs mit einem SDS ist in Tabelle 1 illustriert.

Auch in früheren Versionen von ACT-R wurde ein Stack verwendet, um die Abfolge von Teilzielen bei der Durchführung einer Aufgabe zu organisieren. Dies wurde jedoch von verschiedenen Autoren kritisiert, insbesondere da beim Menschen vorhandene Gedächtniseinschränkungen dabei nicht berücksichtigt werden und die durch den Stack vorgegebene Reihenfolge der Teilziele teilweise nicht mit empirischen Beobachtungen vereinbar ist. Zudem ist menschliches Verhalten oft stärker von der Situation als von einer Handlungsplanung abhängig [13]. Die beiden letzteren Einwände gehen allerdings offenbar davon aus, dass die Planung, und damit die Reihenfolge der Teilziele im Stack, vor Beginn der Aufgabenausführung abgeschlossen ist, was bei dem von Schatzmann et al. vorgestellten Modellierungsansatz nicht der Fall ist, da hier während der Interaktion neue Ziele obenauf gelegt werden können.

Andererseits zeigen eigene Erfahrungen mit dem AGM, dass die Zahl *n* der Aktionen, die zur Bildung einer Benutzeraktion „gepoppt" werden, meistens sinnvoll nur in Abhängigkeit von der jeweiligen Situation bestimmt werden kann. Als Beispiel sei hier die Situation nach dem 5. System-Prompt in Tabelle 1 betrachtet. Hier werden die Konzepte *logical*, *field* und *price* in die Agenda gepusht, und n=1. Da es sich um ein probabilistisches Modell handelt, wären auch andere Varianten mit einer gewissen Wahrscheinlichkeit möglich. Z. B. könnten n=3 oder n=2 Konzepte für die folgende Benutzeräußerung verwendet werden. Schaut man sich die empirische Daten an, stellt sich aber heraus, dass n=3 nur vorkommt, wenn tatsächlich alle drei Konzepte „gepusht" wurden, und nicht nur *logical* und *price*. Die bloße Reihenfolge der Ziele reicht also nicht aus, um deren Relevanz zu einem bestimmten Zeitpunkt sinnvoll zu modellieren.

System	**User**	**Agenda**	**Activation**
	<Aufgabe wird gestellt>	*food=chinesisch* *price=unterePk* *date=Mittwoch* *time=mittag*	*food=chinesisch (0.9)* *price=unterePk (0.8)* *date=Mittwoch (0.7)* *time=mittag (0.6)*
Guten Abend! Herzlich Willkommen […] nach den folgenden Kriterien durchzuführen: der Art der Küche, […] oder nach dessen Öffnungszeiten. Stellen Sie bitte Ihre Frage.	Nach der Art der Küche.	*field=food* food=chinesisch price=unterePk date=Mittwoch time=mittag	field=food (1) food=chinesisch (0.8) price=unterePk (0.72) date=Mittwoch (0.63) time=mittag (0.54)
Sie können zwischen den folgenden Möglichkeiten wählen: italienische, französische, griechische oder chinesische Küche oder keine Präferenz.	Chinesische Küche in der unteren Preisklasse	food=chinesisch price=unterePk date=Mittwoch time=mittag	food=chinesisch (1) price=unterePk (0.65) date=Mittwoch (0.57) time=mittag (0.49)
Sie möchten also in der oberen Preisklasse chinesisch essen gehen?	Nein, in der unteren!	*logical=no* *price=unterePk* date=Mittwoch time=mittag	*logical=no (1)* *price=unterePk (1)* date=Mittwoch (0.51) time=mittag (0.44)
Suchen Sie ein Restaurant für das Mittagessen oder für das Abendessen?	Das Mittagessen am Mittwoch.	*time=mittag* date=Mittwoch	*time=mittag (1)* date=Mittwoch (0.46)
Es gibt kein passendes Restaurant auf Ihre Anfrage. Möchten Sie die Anfrage modifizieren?	Ja	*logical=yes* *field=price* *price=mittlerePk*	*logical=yes (1)* *field=price (1)* *price=mittlerePk (1)*
Sie können die Art der Küche, die Tageszeit […] oder dessen Öffnungszeiten ändern. Geben Sie bitte Ihre Änderung an.	Mittlere Preisklasse	field=price price=mittlerePk	field=price (0.9) price=mittlerePk (0.9)
Hier ist das passende Restaurant auf Ihre Anfrage: …			

Tabelle 1 – Illustration der Funktionsweise des Agenda-basierten Benutzermodells im Vergleich zum aktivierungsbasierten Modell anhand eines Beispieldialogs.

3 Ein aktivierungs-basierter Ansatz zur Benutzermodellierung

Anhand eines Beispiels soll zunächst gezeigt werden, dass sich das AGM direkt in ein aktivierungs-basiertes Modell (AKTM) überführen lässt. Insbesondere kann die Reihenfolge der Teilziele in der Agenda in Form von unterschiedlichen Aktivierungslevels kodiert werden, wobei das obenliegende Ziel die höchste Aktivierung hat. Die Reihenfolge der Teilziele in der Agenda ließe sich dann also durch Ordnung absteigend nach ihrer Aktivierung wiederherstellen. Auch dies ist in Tabelle 1 illustriert (Aktivierungs-Werte in Klammern).

Die Aktivierung ist zunächst definiert als Wert zwischen 0 und 1. Entsprechend dem Vorgehen beim AGM erfolgt eine Neuberechnung der Aktivierung einmal nach der Systemaktion, sowie nach der Benutzeraktion. Nach der Systemaktion wird die Aktivierung der AVPs, die beim AGM „gepusht" würden, auf 1 gesetzt. Gleichzeitig erfahren alle anderen AVPs, die Ziele des Benutzers repräsentieren, eine Dämpfung, indem sie mit dem Faktor 0,9 multipliziert werden. Damit wird eine abnehmende Grundaktivierung des Ziels über die Zeit modelliert. Die Benutzeraktion wird daraufhin aus den *n* AVPs mit der höchsten Aktivierung gebildet. Die Aktivierung dieser AVPs wird daraufhin auf 0 gesetzt, was der „Pop"-Aktion beim AGM entspricht. Um auszuschließen, dass bereits geäußerte AVPs ohne erneute Aktivierung nochmals in einer Benutzeraktion auftauchen, werden bei der Bildung letzterer nur AVPs mit einer Aktivierung größer 0 berücksichtigt.

Das Agenda-basierte Benutzermodell lässt sich also direkt in ein auf Aktivierung von Konzepten basierendes Benutzermodell übersetzen, stellt also prinzipiell die gleiche Expressivität bereit wie ersteres.

3.1 Aktivierungsabhängige Auswahl von Konzepten

Wie oben bereits erwähnt, berücksichtigt die beschriebene Implementierung, die dem AGM entspricht, nicht die für Menschen typische Gedächtnisleistung und ist zu wenig situationsspezifisch im Hinblick auf die in den Benutzeräußerungen auftretenden Konzepte, sowie die Anzahl der Konzepte. Da die Aktivierungswerte neben der Reihenfolge der Konzepte auch die seit der letzten Aktivierung vergangene Zeit speichern, lässt sich die Imperfektion des Gedächtnisses relativ leicht modellieren, indem ein minimaler Aktivierungswert als Voraussetzung für das „poppen" eines Konzepts postuliert wird. Ähnlich könnte man annehmen, dass die Anzahl *n* der Konzepte in der Benutzeräußerung von den Aktivierungswerten abhängt, bzw. dass jeweils alle Konzepte „gepoppt" werden, die einen bestimmten Grenzwert überschreiten.

Dieser Grenzwert muss zunächst vorgegeben werden, und wurde in einer Beispielimplementierung auf 0.9 gesetzt. Würden nun beispielsweise 3 AVPs aktiviert, d.h., ihre Aktivierungswerte auf 1 gesetzt, würden diese unmittelbar in die nächste Benutzeräußerung übernommen werden. Das Modell würde also weniger Varianz in seinem Verhalten zeigen als das AGM. Dieses Verhalten ist unerwünscht, kann jedoch dadurch ausgemerzt werden, dass für jedes AVP, das aktiviert wird, ein zugehöriger Aktivierungswert bestimmt wird. Dieser kann auch als Zufallszahl in einem bestimmten Intervall generiert werden. So führt eine zufällig aus dem Intervall [0,85; 0,95] gewählte Aktivierung zu einer 50-prozentigen Wahrscheinlichkeit, dass das AVP in der nächsten Benutzeräußerung auftaucht. Wie beim AGM gibt es auch Konzepte, die optional aktiviert werden, z. B: das *field*-Konzept im fünften Dialogschritt im Beispiel in Tabelle 1.

Wir kein Konzept durch den vorangehenden Prompt genügend aktiviert, werden die Aktivierungswerte normalisiert, d.h., sie werden mit dem Multiplikativinversen des höchsten Aktivierungswertes multipliziert, so dass die größte vorhandene Aktivierung 1 beträgt.

Zu Beginn des Dialoges würde demnach im oben angeführten Beispiel (Tabelle 1) nur eines der AVPs die nächste Benutzeraktion bilden. Da die Aktivierungen hier aus dem Intervall [0; 1] gesampelt wurden, wäre dies sogar relativ häufig der Fall. In den vorliegenden Interaktionsdaten realer Benutzer zeigt sich jedoch, dass häufig ein Großteil der AVPs direkt in der ersten Äußerung des Benutzers abgehandelt wird. Man kann dies jedoch damit begründen, dass die gesamten AVPs durch das lesen der Aufgabenstellung aktiviert wurden, da es sich um Daten aus einem Labor-Experiment mit vorgegebenen Aufgaben handelt. Entsprechend wurden in der Beispielimplementierung die Aktivierungen aller Konzepte zu Beginn aus dem Intervall [0,9; 1] gesampelt.

Wenig überraschend zeigt sich nun, dass nur in Ausnahmefällen ein im Gedächtnis des Benutzermodells gespeichertes AVP abgerufen wird, da die Aktivierung mit jedem Interaktionsschritt abnimmt und somit nie größer als 0,9 werden kann. Jedoch kann das AVP erneut aktiviert werden, wobei sich 2 Fälle unterscheiden lassen. Im ersten Fall wird das Konzept direkt durch einen System-prompt aktiviert, womit es in den meisten Fällen direkt in die nächste Benutzeräußerung übergeht. Im zweiten Fall wir das Konzept jedoch indirekt dadurch aktiviert, dass ein semantisch verwandtes Konzept aktiviert wurde. So könnte z. B. die direkte Aktivierung des Attributes *date* zu einer Mit-Aktivierung des Attributes *time* führen. Diese Mit-Aktivierung würde in diesem Fall zu der Grundaktivierung des AVPs *time* hinzuaddiert werden. Dieser Effekt ist als „Spreading Activation“ bei der Modellierung des menschlichen Gedächtnisses gut dokumentiert und von daher psychologisch plausibel.

In der Beispiel-Implementierung ist dieser Effekt zunächst sehr einfach umgesetzt. Zunächst werden nur die Attribute betrachtet, und die jeweiligen Werte außer Acht gelassen. Die gegenseitige Aktivierung wird in einer quadratischen Matrix festgelegt, in der die Zeilen und Spalten jeweils den Attributen entsprechen, und die Wert in Zeile *i* und Spalte *j* jeweils die semantische Verwandtschaft zwischen den Attributen *i* und *j* ausdrückt.

Der Effekt der *Spreading Activation* erlaubt also eine erneute, teilweise Aktivierung von AVPs. Ob das mit-aktivierte Konzept in der Benutzeraktion auftaucht, hängt demnach von seiner Grundaktivierung, sowie von der Stärke des Semantischen Zusammenhangs mit dem direkt aktivierten AVP ab.

4 Evaluierung

An dieser Stelle soll eine vorläufige Evaluierung des vorgeschlagenen Modells unternommen werden. Dazu werden simulierte Dialoge mit Dialogen realer Versuchspersonen im Laborexperiment verglichen. Als Beispielsystem wird das BoRIS Restaurantinformations-system verwendet. Es unterstützt die Suche nach Restaurants in einem Gemischte-Initiative-Dialog. Das Beispiel in Tabelle 1 illustriert einen typischen Dialogablauf mit BoRIS.

In Tabelle 2 finden sich verschiedene Maße zum Vergleich der in beiden Korpora generierten Benutzeraktionen. Die Evaluierung wurde zunächst für alle getesteten Aufgaben durchgeführt. Es zeigt sich, dass mit dem AGM mehr unterschiedliche Äußerungen generiert wurden als mit dem AKTM, was zu einer höheren Anzahl von Aktionen, die in beiden Korpora vorhanden sind, führt. Entsprechend ist auch der Wert für *Recall* (Anteil der empirisch beobachteten Äußerungen, der auch simuliert wurde) für AGM höher. Der Vorteil des AKTM liegt jedoch in der *Precision* (Anteil der simulierten Äußerungen, die auch empirisch beobachtet wurden), die hier höher ist als beim AGM. Insgesamt muss gesagt werden, dass insbesondere bei der Evaluierung von Systemen eine hohe Anzahl unterschiedlicher Benutzeraktionen wünschenswert ist. Da das neue Modell jedoch zunächst v. a. realistischer sein soll, ist das verbesserte Ergebnis bei der *Precision* ein zufriedenstellendes Zwischenergebnis.

	N(emp)	*N(sim)*	*N(emp) & N(sim)*	*Recall*	*Precision*
AGM (alle Aufg.)	113	172	63	0.56	0.37
AKTM (alle Aufg.)	113	95	39	0.35	0.41
AGM (Aufg. 2)	40	77	26	0.65	0.34
AKTM (Aufg. 2)	40	27	17	0.43	0.51

Tabelle 2 – Evaluierungsergebnisse für AGM und AKTM bei unterschiedlichen Aufgaben.

Schaut man sich nur die vollständig spezifizierte Aufgabe 2 an (bei den anderen Aufgaben wurden nicht alle Kriterien durch die Aufgabe vorgegeben), verstärkt sich der Effekt der größeren *Precision* beim AKTM noch, während sich der Unterschied beim *Recall* manifestiert. Es mss hier erwähnt werden, dass insbesondere beim AKTM die Behandlung nicht spezifizierter Aufgabenkriterien noch verbessert werden muss. Zur Zeit wird hier der Wert „neutral“ strikt als Kriterium vorgegeben, während reale Benutzer i. d. R. einen konkreten Wert erfanden, wenn das System ein nicht spezifiziertes Attribut abfragte.

5 Diskussion und Schlussfolgerungen

Obwohl das vorgeschlagene Benutzermodell recht plausibel erscheint und in Vorarbeiten, insbesondere im Zusammenhang mit ACT-R und der PSI-Theorie, ähnliche Modellierungsansätze verfolgt wurden, muss ein Nachweis der Angemessenheit des vorgeschlagenen Modells noch erbracht werden. Hinsichtlich der empirischen Validierung reichen Ergebnisse für einen Datensatz sicherlich nicht aus, sondern es müssen Interaktionen mit unterschiedlichen Systemen analysiert werden. Da derart komplexe Hypothesen eigentlich nur noch durch Anwendung des Modells auf empirischen Daten bewertet werden können, wäre ein Verfahren zur maschinellen Bestimmung der Modellparameter hilfreich. Soll das Modell für die Evaluierung von Systemen genutzt werden, ist zudem eine Untersuchung der Komplexität der Modell-Erstellung angezeigt.

Da die Schwierigkeit bei der Benutzermodellierung v.a. darin liegt, sinnvolle Äußerungen, d.h., Kombiantionen von AVPs in einem bestimmten Kontext, zu generieren, scheint das Konzept der *Spreading Activation*, das dieses Verhalten wesentlich beeinflusst, relativ wichtig zu sein. Grundsätzlich ist eine adäquatere Modellierung hier zunächst dadurch zu erreichen, dass statt der Attribute AVPs hinsichtlich ihrer Ähnlichkeit verglichen werden. Allerdings erfordert die Spezifikation einer derart großen Menge semantischer Ähnlichkeiten eine Automatisierung dieser Aufgabe. Zur Bestimmung domänenunabhängiger Wort-Ähnlichkeiten aus großen Sprachkorpora ließe sich prinzipiell ein Tool wie DISCO [14] verwenden.

Möglicherweise lässt sich mit Hilfe der *Spreading Activation* auch erklären, warum Benutzer teilweise nicht direkt aus der Aufgabe hervorgehende Konzepte verwenden. Zu diesen Fällen zählt v. a. die spontane Änderung von Werten für ein bestimmtes Attribut (z. B. „chinesisch“ statt wie in der Aufgabe vorgegeben „Ente“).

Sollte sich das beschriebene Modell als valide herausstellen, wäre als eine erste konzeptionelle Erweiterung eine quasi zeitkontinuierliche Modellierung denkbar. Die Dämpfung vorangegangener AVPs würde dann von der tatsächlich verstrichenen Zeit abhängen, anstatt von der Anzahl der seither erfolgten Dialogschritte. Neben einer genaueren Modellierung der Aktivierungs-Dämpfung in Abhängigkeit von der Zeit, wäre ein Vorteil einer solchen Implementierung, dass prinzipiell die Modellierung von Barge-in (d.h., der Benutzer unterbricht das System und macht seine Eingabe frühzeitig) möglich wäre. Dazu könnte man eine Aktivierungsschwelle zu bestimmen, ab der ein Benutzer tatsächlich nicht mehr das Ende des Prompts abwartet, sondern das System unterbricht.

Ein wesentlicher Grund, eine solche auf kognitiven Prozessen statt statistischen Häufigkeiten beruhende Modellierung vorzuschlagen, war die angestrebte Integration von Modellen des Benutzerverhaltens und des Beurteilens der Interaktion. Wie bereits erwähnt könnte der Weg hier über die Simulation von Affekten und Emotionen gehen, wie sie zum Beispiel in der PSI-Theorie spezifiziert ist. Die PSI-Theorie betrachtet Affekte und Emotionen als Modulation kognitiver Prozesse, zu denen auch die Zielauswahl gehört. Ärger wird demnach z. B. u. a. als Zustand der Fokussierung auf ein bestimmtes Ziel charakterisiert. Hinzu kommen natürlich noch andere Aspekte, wie z. B. eine niedrige Auflösung bei der Wahrnehmung und beim Planen. Grundsätzlich bedeutet die Modellierung von Benutzerverhalten mittels Aktivierung von Teilzielen aber einen Schritt in die Richtung der Simulation emotionalen Verhaltens.

Literatur

[1] Chung, G.: Developing a Flexible Spoken Dialog System Using Simulation, in: Proc. of ACL '04, Barcelona, Spain, 2004.

[2] López-Cózar, R., Callejas, Z., McTear, M.: Testing the Performance of Spoken Dialogue Systems by Means of an Artificially Simulated User. Artificial Intelligence Review 26, 2006, pp. 291-323.

[3] Möller, S., Schleicher, R., Butenkov, D., Engelbrecht, K.-P., Gödde, F., Scheffler, T., Roller, R. and Reithinger, N.: Usability Engineering for Spoken Dialogue Systems Via Statistical User Models. IWSDS 2009, Kloster Irsee, Germany, 2009.

[4] Schatzmann, J., Thomson, B., Weilhammer, K., Ye, H., Young, S.: Agenda-based User Simulation for Bootstrapping a POMDP Dialogue System, in: Proc. of HLT/NAACL, Rochester, NY, USA, 2007.

[5] Pietquin, O.: A Framework for Unsupervised Learning of Dialogue Strategies, Ph.D. thesis, Faculty of Engineering, Mons (TCTS Lab), Belgien, 2004.

[6] Engelbrecht, K.-P.: Estimating Spoken Dialog System Quality with User Models, Dissertation, Fakultät für Elektrotechnik und Informatik, TU Berlin, 2012.

[7] Walker, M., Litman, D., Kamm, C., Abella, A.: PARADISE: A Framework for Evaluating Spoken Dialogue Agents, in: Proc. of the ACL/EACL 35th Annual Meeting of the Association for Computational Linguistics, Madrid, 1997, pp. 271–280.

[8] Norman, D.: Emotional Design: Why we love (or Hate) Everyday Things. Chapter 1: Attractive Things Work Better. Basic Books, New York, NY, USA, 2004.

[9] Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., Qin, Y.: An Integrated Theory of the Mind. Psychological Review 111(4), 2004, p. 1036-1060.

[10] Dörner, D.: Die Mechanik des Seelenwagens. Eine neuronale Theorie der Handlungsregulation. 1. Auflage, Verlag Hans Huber, Bern, Switzerland, 2002.

[11] Altmann, E. M. & Trafton, J. G.: Memory for Goals: An Architectural Perspective. Proceedings of the twenty first annual meeting of the Cognitive Science Society, 1999, pp. 19-24.

[12] Putze, F., Schultz, T.: Utterance Selection for Speech Acts in a Cognitive Tourguide Scenario, Proc. of Interspeech 2010, Makuhari, 2010.

[13] VanLehn, K., & Ball, W.: Goal reconstruction: How Teton blends situated action and planned action. In: VanLehn, K. (Ed.): Architectures for intelligence. Hillsdale, NJ: Erlbaum, 1991, pp. 147–189.

[14] http://www.linguatools.de/disco/disco.html, zuletzt gesichtet am 9. 7. 2012