Fehlerklassifikation und Benutzbarkeits-Vorhersage für Sprachdialogdienste auf der Basis von mentalen Modellen

Magisterarbeit

von

Klaus-Peter Engelbrecht

Fachgebiet Kommunikationswissenschaft

Fakultät I

der Technischen Universität Berlin

angefertigt an den

Deutsche Telekom Laboratories

Technische Universität Berlin

Berlin, im Oktober 2006

Erklärung

Ich erkläre an Eides Statt, dass ich die Magisterarbeit mit dem Titel „Fehlerklassifikation und Benutzbarkeits-Vorhersage für Sprachdialogdienste auf der Basis von mentalen Modellen“ selbständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und alle den benutzten Quellen wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Inhalt

1 Einleitung

Sprachdialogdienste sind aus der modernen Gesellschaft nicht mehr wegzudenken. Ein Sprachdialogdienst führt einen Dialog in gesprochener Sprache mit einem Benutzer, an dessen Ende dem Benutzer ein Service geboten wird. Allgemeiner kann man auch von Sprachdialogsystemen (SDS) sprechen. Technisch gesehen handelt es sich um eine Software, die in die Komponenten Spracheingabeverarbeitung, Dialogmanagement und Sprachgenerierung gegliedert werden kann.

Aufgrund des recht soliden Entwicklungsstands von Sprachdialogsystemen werden diese mittlerweile von vielen Unternehmen benutzt, um insbesondere telefonischen Kundenservice kostengünstig anbieten zu können. Zu den bekanntesten deutschen Anwendungen dieser Art gehört die telefonische Fahrplanauskunft der Deutschen Bahn. Fast ebenso bekannt wie das System selbst sind mittlerweile die Produkte der komödiantischen Unterhaltungsindustrie, in denen abstruse Dialoge mit dem Computer nachgestellt werden, die dem Anrufer letzten Endes in den Wahnsinn treiben, indem sie ihn mit Miss- und Unverständnis malträtieren.

Hier wird im Wesentlichen die Spracherkennungskomponente aufs Korn genommen, und in der Tat ist diese in der Regel das schwächste Glied in der Verarbeitungskette. Mit zunehmender Komplexität der Dienste entstehen jedoch auch größere Ansprüche an das Dialogmanagement und die Sprachgenerierung. Während letztere für das Feedback an den Benutzer verantwortlich ist, diesem also vermitteln muss, was er mit dem bisherigen Dialog erreicht hat und wie er weiter vorzugehen hat, um sein Ziel zu erreichen, muss das Dialogmanagement einerseits einen sinnvollen Ablauf der Akquisition aller nötigen Informationen zur Lösung der Aufgabe gewährleisten, zum anderen aber auch – und hierin besteht die Herausforderung beim Design des Systems – möglichst gut mit einem fehlerhaften Ablauf fertig werden bzw. dem Benutzer ermöglichen, Fehler zu erkennen und zu reparieren.

Eine Ursache von Problemen im Dialogverlauf liegt insbesondere bei komplexeren Systemen oft darin, dass der Benutzer eine andere Vorstellung vom Ablauf des Dialogs

oder den Möglichkeiten des Systems hat als tatsächlich implementiert ist. Die Vorstellung des Benutzers von der Funktionsweise des Systems, bzw. die Strategien zur Lösung einer Aufgabe mit dem System, werden im Bereich der Mensch-Maschine-Interaktion häufig mit dem Begriff der Mentalen Modelle erfasst. Der Begriff wurde aus der Psychologie für dieses Beschäftigungsfeld übernommen und beschreibt im Wesentlichen eine Vorstellung von der kognitiven Repräsentation von Systemen beim Menschen.

Neben wirtschaftlichen Faktoren bildet die Zufriedenheit der Benutzer, die als Konsequenz aus den Faktoren Komfort, Problemlösungseffizienz und Kommunikationseffizienz zum Teil auf Interaktionsprobleme zurückführbar sein sollte, einen wesentlichen Anteil an der Akzeptanz des SDS (Möller 2005b). Deshalb ist die Zufriedenheit eines Benutzers mit dem System von größtem Interesse für Serviceanbieter und Entwickler der Systeme. Um Zeit und Kosten bei der Evaluation der Dienste zu sparen, rückt mehr und mehr die automatische Vorhersage von Qualitätsurteilen aus parametrischen Beschreibungen bereits geführter Dialoge ins Blickfeld der Usability-Forscher.

Ziel der vorliegenden Arbeit ist, Möglichkeiten der Vorhersage von Qualität aus Interaktionsfehlern zu untersuchen. Dafür sollen Fehler in Dialogen mit zwei relativ komplexen SDS analysiert werden. Das erste ist ein sog. Smart-Home-System, mit dem, basierend auf Sprache, Hausgeräte gesteuert werden können. Das System wurde im Rahmen des INSPIRE-Projektes (INfotainment management with SPeech Interaction via REmote microphones and telephone interfaces; EU-gefördertes IST-Projekt IST-2001-32746) aufgebaut. Mit diesem System wurden drei Versuche durchgeführt, zu denen transkribierte Dialoge und Benutzerurteile vorliegen. In dieser Arbeit soll der erste dieser drei Versuche verwendet werden. Das zweite hier betrachtete System ist ein telefonbasiertes System für Restaurantauskünfte (Bochumer Restaurant-Informations-System, BoRIS), welches an der Ruhr-Universität Bochum entwickelt wurde. Mit diesem System wurden in einem Experiment 197 Dialoge geführt und ebenfalls transkribiert. Zudem wurden auch hier für jeden Dialog Benutzerurteile zu verschiedenen Qualitätsaspekten gesammelt.

Zuerst sollen Fehler in den Dialogen mit dem INSPIRE-System ausfindig gemacht und klassifiziert werden. Dabei werden insbesondere Fehler beachtet, die mit dem Mentalen Modell des Benutzers zusammenhängen. Der komplette Datensatz soll annotiert werden, und es sollen Korrelationen der Fehler mit den Benutzerurteilen berechnet und die Ergebnisse interpretiert werden. Auf Grundlage der daraus gewonnenen Informationen soll das Klassifikationsschema optimiert und anschließend für die Annotierung der Daten aus dem Experiment mit BoRIS verwendet werden. Auch hier sollen Korrelationen zu den Benutzerurteilen berechnet und mit den Ergebnissen aus der Analyse des INSPIRE-Systems verglichen werden. Schließlich sollen aus den Fehlerklassen, die Auswirkungen auf die Benutzerurteile zu haben scheinen, Vorhersagen der Benutzerurteile mittels Regressions- und Klassifikationsbäumen versucht werden.

Zunächst sei jedoch ein kurzer Überblick über die Theorie der Mentalen Modelle sowie einige frühere Arbeiten zur Klassifikation von Fehlern gegeben.

2 Mentale Modelle und Fehler

2.1 Mentale Modelle

2.1.1 Mentale Modelle

In seiner auf die Beschäftigung mit Mentalen Modellen in der Mensch-Maschine-Interaktion überaus einflussreichen Schrift von 1983 schreibt Norman:

> *People's views of the world, of themselves, of their own capabilities, and of the tasks that they are asked to perform, or topics they are asked to learn, depend heavily on the conceptualisations that they bring to the task. In interacting with the environment, with others, and with the artifacts of technology, people form internal, mental models of themselves and of the things with which they are interacting.*[1]

Was man sich unter einem Mentalen Modell vorzustellen hat, soll im Folgenden zu erklären versucht werden. Es sollte jedoch keine exakte Beschreibung erwartet werden, da wegen der nur indirekten Möglichkeiten zur Beobachtung mentaler Prozesse keine genaue Definition der Modelle existiert. Stattdessen nähern sich die verschiedenen Forscher theoretisch und empirisch dieser Theorie der Repräsentation von Aufgaben in ihren unterschiedlichen Aspekten und kommen dabei mitunter zu nicht übereinstimmenden Schlussfolgerungen. Dennoch lässt der Forschungsstand einige Aussagen über die Beschaffenheit Mentaler Modelle sowie deren Nutzen für die Lerntheorie zu.

In der Mensch-Maschine-Interaktion werden mit Mentalen Modellen die kognitiven Vorgänge bei der Benutzung relativ komplexer Systeme (z. B. eines einfachen Taschenrechners) beschrieben. Insbesondere ist hierbei interessant, dass auch Neulinge, die ein System noch nie benutzt haben, in der Regel Probleme mit dem Gerät lösen können oder zumindest verschiedene Ansätze dazu entwickeln. Da bei der Erstbenutzung kein deklaratives Wissen vorhanden sein kann, zudem die Funktionsweise des Systems nicht vollständig aus dessen äußerer Form ableitbar ist, müssen hier Vorgehensweisen von

[1] Norman (1983, S. 7)

früheren Erfahrungen übertragen oder aus diesen gefolgert werden. Bei häufigerem Umgang mit dem Gerät baut der Benutzer auf Grundlage seiner Erfahrungen ein Mentales Modell auf, das ihn unter anderem befähigt, fragen über die Funktionsweise des Systems (die er niemals direkt wahrgenommen hat) zu beantworten.

Moray (1999) bemerkte, dass Mentale Modelle in ihrer Wirkungsweise bewusst oder unbewusst sein können. So können Versuchspersonen, die einen Taschenrechner bedienen, über ihre jeweiligen Schritte reflektieren, während sie sie abwägen. Andererseits gibt es auch unbewusste mentale Vorgänge, die den Mentalen Modellen zugeschrieben werden. Z. B. ist das Wissen, dass Kühlwasser, das seine Funktion erfüllt hat, wärmer geworden ist, unmittelbar präsent. Moray vermutet weiterhin Modelle, die unbewusste und bewusste Anteile haben oder vom Bewussten ins Unbewusste übergehen.

Mentale Modelle können ein System auf verschiedenen Betrachtungsebenen repräsentieren. Allen (1997) konstatiert z. B. für die Benutzung eines Computers die Ebenen

- Physisch (Bauteile, Funktionsweise)
- Betriebssystem
- Applikation (Anwendung)
- Software

Diese Ebenen müssen jedoch nicht jedem Benutzer zugänglich sein; so kann für dieses Beispiel bei einem Mikroelektroniker eher ein Mentales Modell auf der physischen Ebene vermutet werden als bei einer Person, die den Computer nur gelegentlich benutzt, um Dokumente zu erstellen. Wie bei Computern kann man auch bei SDS von verschiedenen Ebenen ausgehen. Zu diesen könnte man zählen

- Dialog (Problemlösung z. B.: Welche Information muss ich zuerst geben?)
- SDS an sich (Was mache ich, wenn das System mich nicht versteht?)
- Voice-Server (die Beantwortung der Frage dauert lange, weil gerade viele Leute anrufen.)

Die Definitionen und Beschreibungen Mentaler Modelle geben gewöhnlich keine Auskunft darüber, wie sich der Autor die mentale Repräsentation des Modells vorstellt. So

wäre z. B. eine räumliche oder vokabularische Repräsentation denkbar. Rouse und Morris (1986) halten die bildliche Repräsentation für nahe liegend angesichts der guten Fähigkeiten des Menschen zur Mustererkennung. Sie bemerken zudem, dass (in der Kognitionswissenschaft) die Vorstellungen von Mentalen Modellen von sehr allgemeinen zu sehr speziellen Konstrukten reichen. Relativ einig ist man sich jedoch, dass Mentale Modelle lückenhaft und sogar inkonsistent sind. Gentner und Gentner (1983) wiesen darauf hin, dass einige, aber nicht alle Eigenschaften des realen Systems in die Modelle eingeschlossen sind.[2] Insofern deckt sich die Definition mit der generellen Definition von Modellen, die besagt, dass im Modell die relevanten Eigenschaften für eine bestimmte Fragestellung kondensiert sind.

In der Literatur zum Thema wurde auch diskutiert, wie sich Mentale Modelle von der Repräsentationsform des deklarativen Wissens abgrenzen, die in gewissem Sinne eine konkurrierende Theorie darstellt. Die Idee der Mentalen Modelle ist wohl, dass diese über das deklarative Wissen hinaus auch dynamische Beziehungen zwischen Elementen repräsentieren.
Rouse und Morris (1986), die eine Definition der Modelle über ihren Nutzen versuchen, bemerken, dass ihre Definition nicht ausschließt, dass das Modell durch deklaratives Wissen repräsentiert wird, stattdessen werde ein Bereich des Wissens und die Zwecke, die dieses erfüllt, spezifiziert. Moray (1999) dagegen versucht seine Vorstellung davon, inwiefern deklaratives Wissen eben doch überschritten wird, zu vermitteln, indem er anmerkt, man habe z. B. ein Modell von der Frau, die man liebt.

Da es sich bei Mentalen Modellen im hier interessierenden Zusammenhang gewöhnlich um kognitive Beschreibungen von Handlungs- bzw. Funktionsweisen handelt, die bei einer konkreten Handlung zutage treten, stellt sich auch die Frage nach der Dauerhaftigkeit der Modelle bzw. dem mentalen Ablauf der Benutzung eines solchen kognitiven Konstrukts. Johnson-Laird, Byrne and Schaeken (1992) vertreten diesbezüglich die Meinung, es handele sich um eine Repräsentation des Inhalts eines Problems im Kurzzeitgedächtnis, was die Anwendung von Problemlösungsstrategien darauf

[2] Wiedergegeben nach Allen (1997).

ermögliche.[3] Bei Beschäftigung mit der Mensch-Maschine-Interaktion wird der Begriff hingegen laut Moray (1999) auf das Langzeitgedächtnis bezogen (der Operateur hat ein mentales Modell von der Maschine, die er bedient). Das Modell ist Ergebnis seiner Ausbildung und Erfahrung im Umgang mit der Maschine und kann genutzt werden, um die Interaktion mit der Maschine zu bestimmen. Moray schließt jedoch nicht aus, dass ein kurzzeitiges Modell für nur ein Teilproblem (mit) der Maschine erstellt wird

Aus dem zuletzt Gesagten wird deutlich, dass Mentale Modelle von ein und demselben System sich bei verschiedenen Menschen unterscheiden. Insbesondere haben routinierte Benutzer eines Systems (Experten) offensichtlich ein anderes (besseres) Modell von diesem als Neulinge. Nach Rouse und Morris (1986) sind die Expertenmodelle jedoch nicht einfach elaborierter als die von Neulingen. Stattdessen vermuten sie bei Experten eine bessere Organisation des Wissens (nach Larkin (1983)) oder einen höheren Abstraktheitsgrad (und damit ein breiteres Anwendungsfeld; nach Glaser (1985) und Pennington (1985)) des Modells. Dabei, so die Autoren, bleiben manchmal die Modelle des Neulings auch erhalten, wenn dieser schon als Experte betrachtet werden kann und das entsprechende Modell bereits „berichtigt“ wurde (nach DiSessa (1982), McCloskey (1983)). Dies kann daran liegen, dass die Alltagserfahrung das Modell nie in Frage stellt, sondern eher noch bestätigt. Dass sich Novizen zu Experten entwickeln können, impliziert schließlich, dass sich ein Mentales Modell verändern kann, wenn durch den Umgang mit einem System oder eine theoretische Einweisung in dessen Funktionsweise neue Erfahrungen bzw. Wissen hinzu kommen.

Auf Grundlage der Zusammenfassung verschiedener Texte zum Nutzen Mentaler Modelle durch Rouse und Morris (1986) soll an dieser Stelle etwas detaillierter auf diesen eingegangen werden. Die Autoren finden eine gemeinsame Sicht aller Texte darin, dass die Modelle Beschreibungen des Systems, Erklärungen der Vorgänge und die Vorhersage eines späteren Zustands des Systems ermöglichen. Diese Erkenntnis wird von den Autoren mit einer Taxonomie der Mentalen Modelle nach Rasmussen (1979) kombiniert. Diese beschreibt vier Betrachtungsebenen von mentalen Modellen, nämlich den Zweck (was ein System macht), die Funktion (wie ein System arbeitet), den Zustand (was das

[3] Wiedergegeben in Moray (1999).

System gerade tut) und die Form (wie das System aussieht). Mentale Modelle, so Rouse und Morris, ermöglichen demnach dem Menschen die Beschreibung von Zweck und Form eines Systems, die Erklärung von Funktion und Zustand sowie die Vorhersage des Zustands.

2.1.2 Mentale Modelle identifizieren

Ein fundamentales Problem bei der Erforschung Mentaler Modelle besteht darin, dass sie sich nicht direkt beobachten lassen und bei einer indirekten Beobachtung darauf geachtet werden muss, dass durch die Beobachtungssituation das Modell nicht verfälscht wird. Mit indirekter Beobachtung ist hier das Ableiten von Einsichten in Form oder Struktur des Modells anhand empirischer Studien gemeint, die z. B. in der Beobachtung des Verhaltens einer Versuchsperson oder einer verbalen Beschreibung der inneren Prozesse durch den Probanden selbst bestehen können. Zudem können Mentale Modelle des selben Systems bei verschiedenen Menschen wie gesagt unterschiedlich sein – u. a. auch bezüglich des „Mappings" des realen Systems auf das Modell (Moray 1999) – und verändern sich ständig bzw. passen sich an. Rouse und Morris (1986) kommen deshalb zu dem Schluss, die Möglichkeiten, Mentale Modelle komplett zu erfassen, seien „rather remote"[4].

Wie Norman (1983) feststellte, hat der Wissenschaftler, der ein Mentales Modell untersucht, wiederum nur ein Mentales Modell von seinem Untersuchungsgegenstand. Norman nennt das Modell des Wissenschaftlers „Conceptual Model". Der beschriebene Umstand macht eine weitere Schwierigkeit bei der Identifizierung Mentaler Modelle deutlich, zumal das konzeptualisierte Modell auch vom methodologischen Hintergrund des Forschers und seinen Erwartungen davon, wie Menschen vermutlich mit dem betrachteten System umgehen, geprägt ist (Rouse und Morris 1986).

Für die Identifizierung Mentaler Modelle stehen im Prinzip zwei Arten von empirischen Methoden zur Verfügung: entweder gibt eine Versuchsperson verbale Erklärungen ab, oder eine Vorhersage des Verhaltens der Versuchsperson wird getestet.

[4] Rouse and Morris (1986), S. 353.

Eine gängige Methode der ersten Art ist das verbale Protokoll, bei dem der Proband während oder direkt nach der Durchführung der Aufgabe seine Gedankengänge mitteilt. Die Analyse kann mit Videoaufzeichnungen unterstützt werden. Der Haupteinwand gegen diese Methode ist, dass der Benutzer sich durch seine Erklärung ein evtl. normalerweise unbewusstes Verhalten bewusst macht und sich dadurch sein Verhalten verändert (Moray 1999). Zu dem Verfahren ist weiter zu bemerken, dass es direkt nur Aufschluss darüber gibt, über was die Person nachdenkt, aber nicht darüber, wie sie dies tut (Rouse und Morris 1986). Alternativ kann man die Versuchsperson ihr Verhalten oder das Verhalten des Systems erklären oder vorhersagen lassen (Allen 1997). Wie Allen (1997) bemerkt, sind Fehler bei der Vorhersage meistens am aussagekräftigsten. Hier wird der Zusammenhang zwischen Fehlern und Mentalen Modellen deutlich: Bei der Entscheidung für oder gegen eine Handlung am System wird eine Vorhersage bezüglich des Effekts der Handlung aufgrund des Mentalen Modells des Benutzers getroffen. Ist die Systemreaktion nicht dem Ziel des Benutzers entsprechend, kann man davon ausgehen, dass die Vorhersage fehlerhaft war und damit, dass das Mentale Modell nicht mit dem tatsächlichen System übereinstimmt.

Für den zweiten Methodenbereich ist ein Beispiel das „Analytical Modelling“, bei dem anhand vorhandener Theorien und Daten Annahmen über Form, Struktur und evtl. Parameter von Mentalen Modellen für bestimmte Aufgaben formuliert werden. Auf den Ergebnissen basierend werden Vorhersagen für menschliches Verhalten getroffen und anschließend mit dem tatsächlichen Verhalten verglichen. Obwohl das formulierte Mentale Modell, selbst wenn es eine korrekte Prädiktion erlaubt, nicht mit Sicherheit das tatsächliche Modell des Benutzers ist, taugt es immerhin für Vorhersagen (Rouse und Morris 1986).

Einen ähnlichen Ansatz wählte Sanderson (u. a. 1990), der seine Annahmen jedoch nicht allein auf theoretische Überlegungen, sondern auch auf die Beobachtung vorheriger Handlungen der Testperson gründete. Dafür ließ er die Probanden elektrische Schaltkreise analysierten und schätzte an verschiedenen Punkten das bisherige Wissen und berechnete dann über ein Computermodell, welche nächsten Aktionen schwierig und welche einfach

sein müssten.[5] Für eine ähnliche Methode, bei der jedoch das Wissen des Benutzers besser abgeschätzt werden kann, wird dieser für die zu lösende Aufgabe trainiert. Zum Beispiel könnte man den Probanden gezielt auf ein bestimmtes Mentales Modell zu trainieren versuchen und ihm dann Aufgaben geben, die er erwartungsgemäß gut oder schlecht durchführen kann (Moray 1999).

2.1.3 Nutzungsbereiche Mentaler Modelle

Die Konzeption der Mentalen Modelle wird bisher vor allem in Theorien zum Lernen benutzt, die sich damit beschäftigen, die Ausbildung von Menschen für bestimmte Aufgabenbereiche effektiver und schneller zu gestalten. Dabei sollte nicht davon ausgegangen werden, dass das Wissen über Prinzipien und Funktionsweise eines Systems die Leistung eines Operateurs in jedem Fall verbessert. Morris und Rouse (1985) haben gezeigt, dass diese sich sogar verschlechtern kann. Sie folgern daraus, dass es einer Anleitung zur richtigen Nutzung des Wissens bedarf[6]. Moray (1999) stellt fest, dass theoretisches Wissen über eine zuvor praktisch erlernte Tätigkeit nicht die Leistung eines Operateurs verbessert, wenn dieser keinen Zusammenhang zwischen den beiden verschiedenen Repräsentationen der selben Tätigkeit herstellen kann.

In einem aktuellen Forschungsprojekt der Deutschen Telekom sollen Mentale Modelle genutzt werden, um das Verhalten von Benutzern im Umgang mit Interaktiven Systemen, also z. B. Sprachdialogsystemen, zu simulieren. Auf Basis von simulierten Dialogen sollen grundlegende konzeptionelle Fehler im Dialogdesign und Usability-Vorhersagen getroffen werden. Diese Arbeit leistet einen Teil der Vorarbeit für das Projekt, indem sie Fehler als die phänotypischen Indikatoren von Mentalen Modellen ausfindig macht und klassifiziert.

[5] Wiedergegeben in Moray (1999).
[6] Wiedergegeben in Rouse und Morris (1986).

2.2 Fehler

2.2.1 Terminologie

Der Begriff Fehler wird hier verwendet als Übersetzung des englischsprachigen Begriffs „Error“. Diese Übersetzung ist nicht völlig zureichend, da der Begriff im Englischen eine weitergehende Bedeutung aufweist. Die Übersetzung nach www.leo.org zum Beispiel bietet die weiteren deutschen Wörter Irrtum, Abweg, Fehltritt, Irrung, Messabweichung und Störung. Im Zusammenhang des Untersuchungsgegenstandes sind vor allem die Übersetzungen als „Irrtum“ und „Abweg“ relevant. Auf viele Situationen, die bei der Klassifikation später betrachtet werden, treffen diese Termini eher zu als „Fehler“; so erscheint es zum Beispiel eher unangebracht, von einem Fehler zu sprechen, wenn der Benutzer eine Vokabel benutzt, die das System nicht semantisch interpretieren kann. Besser scheint es hier, von einem Irrtum zu sprechen: der Benutzer geht davon aus, dass das System die Vokabel verstehen kann, was aber nicht der Wahrheit entspricht.
Dennoch soll der Einfachheit halber der Begriff „Fehler“ als Bezeichner für den gesamten Komplex der Unzulänglichkeiten bei der Interaktion zwischen Benutzer und Sprachdialogsystem gebraucht werden. Über eine entsprechende Unterscheidung der „Fehler“ in Irrtümer, Abwege und Fehler im engeren Sinne kann bei weiterer Entwicklung der hier beschriebenen Methode noch einmal nachgedacht werden, zum gegenwärtigen Zeitpunkt würde sie sich aber auf Begrifflichkeiten beschränken und ist deshalb verzichtbar.

2.2.2 Bisherige Arbeiten zur Klassifikation von Fehlern in der Mensch-Maschine-Interaktion

Zur Thematik von Fehlern im Bereich der Mensch-Maschine-Interaktion existiert bereits einige Literatur, die sich jedoch in der Regel mit anderen Eingabemodalitäten als Sprache, also z. B. Tastatur- oder Schaltereingabe befasst (Eine Ausnahme stellt z. B. Constantinides und Rudnicky (1999) dar). Da die dabei auftretenden Fehler nicht ohne weiteres mit denen bei Sprachdialogdiensten vergleichbar sind, können dieser Vorarbeit lediglich Klassifikationen eher genereller Natur sowie einige Denkanstöße entnommen

werden. An dieser Stelle sei auf zwei Texte aufmerksam gemacht, die das in dieser Arbeit vorgestellte Fehlerklassifikationsschema in dieser Weise beeinflusst haben.

Die erste dieser Schriften stammt von Prabhu und Prabhu (1997) und enthält einige grundlegende Bemerkungen zu Fehlern aus verschiedenen Blickwinkeln. Zuerst wird hier auf die konzeptuelle Unterscheidung zwischen Phänotypen und Genotypen von Fehlern aufmerksam gemacht, die sie auf E. Hollnagel (1989) zurückführen. Mit dem Phänotyp ist hier das äußere Erscheinungsbild des Fehlers gemeint, während als Genotypen von Fehlern deren Ursachen bezeichnet werden, also zum Beispiel die kognitiven Prozesse des Benutzers (die u. a. durch Mentale Modelle beschrieben werden können). Bei der Klassifikation von Fehlern sind also zwei unterschiedliche Ansätze möglich: Klassifikation nach Phänotypen oder nach Genotypen.

Während die Klassifikation von Phänotypen im Wesentlichen auf der Beobachtung (und Zusammenfassung dieser zu Klassen) beruht, ist für die Klassifikation von Genotypen ein gewisser Grad an Wissen bzw. Konzeption der kognitiven Prozesse des Menschen bzw. der Funktionsweise der Maschine nötig, da Genotypen nicht einfach aus dem Dialog ersichtlich sind. Ist das Verhalten einer Maschine prinzipiell aus technischen Details (dargestellt z. B. in einem Blockdiagramm der Funktionsweise) relativ leicht folgerbar, so stößt die Vorhersage *menschlichen* Verhaltens an ihre Grenzen, wo die inneren Prozesse noch nicht ausreichend erforscht sind.

Prabhu und Prabhu stellen in dem genannten Text einige Ansätze für Klassifikationen nach den mentalen Prozessen vor. Dabei unterscheiden sie zunächst zwischen „Slips“ („Flüchtigkeitsfehler“) und „Mistakes“ (konzeptionelle Fehler). Während Slips bei der Ausführung einer geplanten Aktion auftreten, sind mit Mistakes Fehler bei der Planung der Aktion gemeint (z. B. Norman 1981[7]). Dieser Ansatz wurde von Reason (1990) mit einer weiteren Beobachtung von Rasmussen (1986) kombiniert, die besagt, dass sich menschliches Verhalten in drei Stufen aufteilen lasse. Auf der ersten Stufe steht „Skill-based Behaviour“, unbewusstes, sensomotorisches Verhalten mit automatisierten Handlungsmustern, wie z. B. Tastaturtippen oder die Bedienung der Computermaus. Auf

[7] Wiedergegeben in Prabhu und Prabhu (1997).

der zweiten Stufe steht „Rule-based Behaviour“ (regelbasiertes Verhalten), das durch die Anwendung bekannter Regeln gekennzeichnet ist. Ein Beispiel wäre das Öffnen eines Dokuments mit einem bekannten Textverarbeitungsprogramm. Auf der dritten Stufe schließlich steht „Knowledge-based Behaviour“, das in unbekannten Situationen benutzt wird, für die noch keine Regeln existieren. Reason stellte nun fest, dass auf der „Skill-based“ Stufe Slips, also Flüchtigkeitsfehler auftreten, während auf der regel- und wissensbasierten Stufe Mistakes auftreten.

Prabhu und Prabhu weisen auch darauf hin, dass Fehler nicht definiert werden können, ohne dass die Ziele und Erwartungen des Handelnden in Betracht gezogen werden. So kann z. B. die Zuordnung eines Fehlers zur Kategorie Skill-based Slip oder Rule-based Mistake davon abhängen, ob der Benutzer die Handlung „mit Absicht“ getätigt hat oder nicht.

Die zweite Schrift, die hier vorgestellt werden soll, stammt aus der Feder von Lang et al. (1991). Diese untersuchten anhand von 16 Testpersonen deren Umgang mit einer ihnen unbekannten Software. Dabei wurde sowohl eine beschreibende als auch eine erklärende Analyse der Fehler vorgenommen (oder auch: phänotypische und genotypische). Neben Klassifikationsschemen aus früheren Arbeiten wurde dazu auch ein „Ad-hoc“-Schema entworfen. Die Klassen wurden dabei teilweise durch genotypische, teilweise auch durch phänotypische Eigenschaften der Fehler definiert, wie die folgende Übersicht zeigt.

1) Affordance Errors. Features of the object convey how to use it. Alternatively features are missing in the object which would convey how to use it.

2) Message Misinterpretation Errors. User reads a message and does not construe the correct meaning of the message.

3) Goal Induced Errors. User types in a command directly relevant to the goal.

4) Option Identification Error (Menu Option Error). User thinks a menu option performs one action and it performs another.

5) Status Acquisition. User types in commands (usually at the system level) in order to identify the status of the computer.

6) Incomplete procedure. User begins the proper sequence of actions, but "escapes" out before completing the procedure.

7) Pre-requisite action not performed. User fails to perform necessary pre-requisite actions for the error command.

8) Generic Commands. User performs commands which are generally known to achieve a desired function but which failed to achieve this function in the target system (e.g. F1 for Help)

9) Mode errors. User performs an action when the system is in the wrong mode

10) Superstitious. User performs an action for no apparent reason. Action seems to be unexplainable.

11) Errors of omission. User fails to perform a necessary action.

12) Errors of commission. User performs an unnecessary action

Error sequences

Perseveration Error. User performs a sequence of the same error.

Modification sequence. User performs a sequence of errors which are related to each other[8]

[8] Lang et al. (1991), S. 44.

Bei der deskriptiven Anayse wird außerdem nach “display driven errors” und “user initiated errors” unterschieden. „Display driven“ bedeutet, dass der Benutzer die auf dem Display dargebotene Information fehlinterpretiert und dadurch einen Fehler produziert. Bei einem „User initiated Error“ hingegen generiert der Benutzer eine falsche Handlung auf Grundlage seines Vorwissens (z. B. aus Erfahrungen mit ähnlicher Software).

Es sei zu diesem Schema angemerkt, dass die mit „Error Sequences“ überschriebenen Klassen mglw. hilfreich für die Identifikation Mentaler Modelle sind, da häufig auftretende Fehler der selben Art auf eine Misskonzeption des Benutzers schließen lassen Auch einige der weiteren Fehlerkategorien werden nützlich für die Untersuchung von Sprachdialogsystemen wie dem INSPIRE- und BoRIS-System sein, jedoch sind insbesondere jene Klassen, welche auf Genotypen aufbauen, als problematisch für die geplante Analyse zu betrachten, wie später noch erläutert werden wird. Zudem fehlen dieser Klassifikation offensichtlich Fehler, die mit der Spracheingabe zusammenhängen (z. B. Vokabularfehler)

3 Fehlerklassifikation für das INSPIRE-Smart-Home-System

3.1 Datenerhebung

3.1.1 Beschreibung des INSPIRE-Systems

Das INSPIRE-Sprachdialogsystem wurde im Rahmen eines EU-geförderten Projektes u. a. an der Ruhr Universität Bochum entwickelt. Der Name „INSPIRE“ ist ein Akronym des Projektnamens “INfotainment management with SPeech Interaction via REmote-microphones and Telefone interfaces.“ Es handelt sich dabei um ein so genanntes Smart-Home-System, das heißt, man kann es benutzen, um elektronische Hausgeräte wie den Fernseher zu bedienen. Dies geschieht im Wesentlichen über Sprachsteuerung (einige Systemausgaben werden auf einem Bildschirm dargestellt). Dazu führt das System mit dem Benutzer einen Dialog, in dessen Verlauf alle zur Ausführung einer Aufgabe benötigten Informationen gesammelt werden. Eine Besonderheit des INSPIRE-Systems stellt die gemischte Dialoginitiative („Mixed Initiative“) dar. Dies bedeutet, dass der Benutzer die Führung des Dialogs übernehmen kann, indem er Informationen spezifiziert, die vom System nicht direkt erfragt wurden. Die Funktionsweise des Systems soll hier zu Beginn erläutert werden, um einen Einblick in die Anforderungen und Probleme einer Klassifikation von Interaktionsfehlern zu eröffnen.

Über den Dialog lassen sich mit dem System drei verschiedene Lampen, ein Ventilator, ein Fernseher, Rollläden und ein Anrufbeantworter steuern. Die Lampen können über ihr Aussehen (z. B. „gelbe Stehlampe“) oder ihren Standort einzeln angewählt werden. Alle Lampen sowie der Ventilator können ein- und ausgeschaltet werden. Zwei der Lampen sind zusätzlich dimmbar. Der Fernseher lässt sich ein- und ausschalten, und die Lautstärke lässt sich verändern. Zudem ist er mit einer sog. Elektronischen Programminformation (Electronic Program Guide, EPG) ausgestattet. Diese ermöglicht die gezielte Auswahl einer Sendung, indem verschiedene Daten wie Tag, Zeit und Programmart spezifiziert werden. Diese kann dann aufgenommen, gezeigt oder ihr Beginn signalisiert werden. Die Rollläden können geöffnet und geschlossen werden, wobei teilweises Öffnen oder

Schließen über das Kommando „Stopp“ möglich ist. Der Anrufbeantworter schließlich weist die üblichen Funktionen wie Abrufen und Speichern der Nachrichten, sowie Springen zur nächsten Nachricht auf. Die Bedienung der einzelnen Geräte ist also unterschiedlich komplex und bereitet Schwierigkeiten unterschiedlicher Art.

Bei der Implementierung des Systems wurde eine Dialogmanagement-Strategie entworfen, die allgemein für alle verwendeten Geräte und Aufgaben gültig, d. h., von diesen unabhängig ist. Die Aufgaben, oder auch „Lösungen“, werden durch verschiedene Attribute, wie z. B. Gerät, Funktion oder Zeit, bzw. die entsprechenden Werte für diese Attribute (Attribute-Value-Pairs, AVPs) komplett beschrieben. Alle möglichen Lösungen werden als Zeilen in einer Tabelle zusammengefasst, deren Spalten die zur Verfügung stehenden Attribute repräsentieren (Solution Table). Wird ein Attribut (also eine Spalte) für die Beschreibung einer Aufgabe nicht gebraucht, findet sich in der entsprechenden Zeile kein Wert für dieses. Das allgemeingültige Dialogmanagement nun eröffnet den Dialog mit einer offenen Frage („Was kann ich für Sie tun?“). Es nimmt die Eingabe des Benutzers an, extrahiert die der Äußerung entsprechenden AVPs und bestimmt das nächste Attribut, das zur eindeutigen Identifizierung einer Lösung notwendig ist. Nach diesem wird dann gezielt gefragt, der Benutzer kann jedoch auch andere Attributwerte angeben und damit die Initiative im Gespräch übernehmen.

Eine Ausnahme bildet hier die Abfrage des Sendungstitels bei der Benutzung des Fernsehers. Da es sich hierbei um dynamische Attributwerte handelt, d. h., dass diese sich mit der Zeit verändern, würden dabei für das System Probleme mit der Sprachverarbeitung auftauchen. Diese werden jedoch umgangen, indem die Systemausgabe visuell erfolgt (was ohnehin für den Benutzer leichter zu verarbeiten ist) und die Eingabe durch eine Zahl, die dem Titel zugeordnet ist, erfolgt.

Beim Umgang des Systems mit Problemen im Dialog kann man zwischen *lokalen* und *globalen* Mechanismen unterscheiden. Das lokale Dialogmanagement, mittels dessen die Bestimmung eines Wertes für ein bestimmtes Attribut erfolgt, wird als Generic Dialogue Node (GDN) bezeichnet. Darin sind Strategien festgelegt für den Fall, dass das System nicht in der Lage ist, ein AVP vom Benutzer aufzunehmen. Diese Strategien sind im

Prinzip für alle GDNs gültig, jedoch werden entsprechend dem GDN zugehörigen Attribut spezielle Metakommunikations-Prompts festgelegt (als Prompt wird hier eine „Systemäußerung“ bezeichnet). Ein GDN kann mit vier Problemen umgehen:

- Sagt der Benutzer gar nichts (No Input), wird er nochmals gebeten, die aktuelle Frage zu beantworten oder lauter zu sprechen.
- Wurde die Äußerung entweder phonetisch oder semantisch nicht verstanden (No Match), wird die Frage paraphrasiert.
- Weiterhin kann ein Hilfe-Prompt abgespielt werden, wenn der Benutzer danach fragt. Der Hilfe prompt gibt Auskunft über die im aktuellen GDN adäquaten Antwortmöglichkeiten.
- Schließlich kann auf Anfrage des Benutzers die aktuelle System-Frage wiederholt werden.

Wurde ein AVP gefunden, wertet das globale Dialogmanagement die bisher zusammengekommenen AVPs hinsichtlich einer möglichen Lösung aus. Sofern es Lösungen gibt, auf die alle gesammelten AVPs zutreffen, wird das nächste Attribut, für das noch unterschiedliche Werte möglich sind, als nächstes abgefragt. Attribute, für die nur ein Wert möglich ist, werden automatisch mit diesem belegt. Kann mit den bisher gesammelten AVPs jedoch keine Lösung gefunden werden, wird eine Prozedur zur nochmaligen Prüfung eingeleitet, d. h., das System bittet den Benutzer für ein Attribut nach dem anderen, den angegebenen Wert zu überdenken, bis dieser eines ändert („Reconsideration GDN“). Bestätigt der Benutzer alle gesammelten AVPs, wird der Dialog zurückgesetzt, d. h., neu gestartet. Weiterhin kann es passieren, dass für ein Attribut zwei verschiedene Werte gefunden werden. In diesem Fall wird der Benutzer gefragt, welcher Wert der richtige sei. Schließlich ist es bei bestimmten Aufgaben notwendig, dass der Benutzer die gesammelten AVPs vor der Ausführung durch das System bestätigt („Confirmation GDN“). Sind nur noch eine festgelegte Anzahl von Lösungen mit der aktuellen AVP-Konstellation möglich, werden die Möglichkeiten auf dem Bildschirm präsentiert und der Benutzer wird gebeten, durch Sagen der zugehörigen Nummer eine davon auszuwählen.

Die Attribut-Wert-Paare werden vom INSPIRE-System aus natürlichsprachlichen Benutzeräußerungen extrahiert. Die Regeln dafür sind in einer XML-Datei („MappingNLU“) festgelegt, die Stichworte und Phrasen jenen AVPs zuordnet, die die Bedeutung der Aussage für das System repräsentieren. Es wird also keine Wahrscheinlichkeit für eine bestimmte Bedeutung berechnet, wie das bei N-gram-Grammatiken teilweise möglich ist. Mit anderen Worten, nur Phrasen, die wörtlich in dieser Datei enthalten sind, können von dem System verstanden werden, wobei ihre syntagmatische Umgebung nicht berücksichtigt wird. Das bedeutet, dass der Entwickler alle möglichen Äußerung vorhersehen und in der XML-Datei auflisten muss. Gemappt werden sowohl einzelne Wörter als auch Kombinationen von Wörtern (z. B.: „Anrufbeantworter abfragen“ oder „schalte…ein“). Für eine initiale Version der Datei wurden diese Phrasen und Stichwörter „per Hand“ gesammelt, bzw. es wurden Synonyme gesucht, die in einer Antwort auf die Systemfragen auftauchen könnten.

Um eine möglichst umfangreiche Sammlung von Stichwörtern und Phrasen zu bekommen, sowie um verschiedene Systemvarianten zu testen, wurden mit dem INSPIRE-System sog. „Wizard-of-Oz-Experimente“ durchgeführt. Bei diesen Experimenten wird ein noch nicht zur Reife gekommener Systemteil durch einen Menschen (den „Wizard“) ersetzt, so dass die übrigen Systemkomponenten bereits vor Perfektionierung dieses Teils unter realitätsnahen Bedingungen getestet werden können. Im Fall des in dieser Arbeit beschriebenen Experiments mit dem INSPIRE-System wurde die automatische Spracherkennung von einem Menschen übernommen. Zu diesem Zweck wurde für das System eine Benutzeroberfläche entwickelt, in die sowohl die natürlichsprachliche Äußerung der Testperson als auch die kanonischen Werte durch den Wizard manuell eingegeben werden können. Weiterhin stehen dem Wizard verschiedene Kontroll-Interfaces zur Verfügung, die im hier beschriebenen Kontext aber nicht von Bedeutung sind. Das hier analysierte Experiment zeigt das INSPIRE-System in einem sehr frühen Entwicklungsstadium: es war der erste Test mit externen Versuchspersonen. So war z. B. die MappingNLU-Datei noch nicht sehr elaboriert, weswegen insbesondere in diesem Systemteil relativ viele Fehler auftraten.

In diesem Zusammenhang soll noch angesprochen werden, dass das Konzept der Semantikextraktion aus Stichworten, insbesondere in Kombination mit der „mixed initiative“ Eigenschaft prinzipiell Schwierigkeiten mit Homonymen bereitet. So ist zum Beispiel im Fall von INSPIRE das Wort „Nachrichten“ sowohl für die Fernsehnachrichten als auch für die Anrufbeantworternachrichten sinnvoll. Das Wort kann also entweder dem Attribut „TVShowType“ (für die Fernsehnachrichten) oder dem Attribut „Aktion“ (für den AB) zugeordnet werden. Da das System jedoch nicht immer konkret nach einem bestimmten Attribut fragt, geht die Information darüber, welches Attribut das richtige sei, lediglich aus dem Kontext der Äußerung bzw. des Stichwortes hervor. Dieser wird vom System jedoch ignoriert. Infolge dessen muss der Gebrauch homonymer Vokabeln eingeschränkt werden und der Benutzer auf eine Ersatzvokabel (in diesem Falle „Fernsehnachrichten“) gelenkt werden.[9] Hier deutet sich bereits eine Quelle möglicher Interaktionsfehler an.

3.1.2 Versuchsbeschreibung

Der in dieser Arbeit untersuchte Datensatz wurde im Rahmen des INSPIRE-Projektes in einem „Free-Wizard-of-Oz“ genannten Test mit Versuchspersonen erstellt. Dieser wurde im Januar 2004 am Institut für Kommunikationsakustik (IKA) der Ruhr-Universität Bochum durchgeführt.

Das unmittelbare Ziel des Experiments war nicht, Untersuchungen über die Interaktionsfehler anzustellen, sondern eine Abschätzung der Gebrauchstauglichkeit des Systems zu bekommen. So bestand ein erklärter Zweck darin, zu untersuchen, ob ein Benutzer bei Erstgebrauch des Systems in der Lage ist, die Geräte (Lampen etc.) entsprechend eines vorher definierten Szenarios zu bedienen. Weiterhin, ob das System für die Steuerung der vorgesehenen Geräte tauglich ist. Zudem sollten drei verschiedene Interaktionsmetaphern bzgl. ihrer Benutzerfreundlichkeit verglichen werden:

- „Ghost“-Metapher (die Stimme hat kein physisches Pendant)
- „Avatar“-Metapher (auf einem Bildschirm wird eine „sprechende“ Handpuppe dargestellt)

[9] Das selbe Problem gilt natürlich für Homophone, sofern ein automatischer Spracherkenner benutzt wird.

- „Intelligent Devices“-Metapher (die Stimme kommt aus Richtung des Gerätes, das gerade bedient wird)

Mit Hilfe der Testergebnisse sollten die Aufgabenstruktur und die Dialogstrategie weiter verbessert werden. Ein weiteres Ziel des Experiments bestand darin, die Untersuchungsmethode selbst bzgl. ihrer Validität und Reliabilität zu testen und für spätere Analysen einen Datensatz mit Interaktionsparametern und dazugehörigen Benutzerurteilen zu erstellen. Dazu wurde eine Vielzahl von automatisch messbaren Parametern wie Dialogdauer oder Anzahl der Hilfe-Prompts aufgezeichnet.

Der Test wurde in einem speziellen Versuchsraum durchgeführt, um von den Vorteilen von Laboruntersuchungen, wie z. B. bessere Messgenauigkeit, profitieren zu können. Um dennoch eine möglichst realistische Gebrauchssituation zu simulieren, wurde das Labor im Stil eines Wohnzimmers gestaltet. Der 3 x 5 Meter große Raum wurde dafür mit Sofa, Sesseln, einem niedrigen Tisch und Regalen bestückt. In dieser Umgebung wurden die (ebenfalls wohnzimmertypischen) Geräte des INSPIRE-Systems platziert. Für die Darstellung des Avatars wurde ein zusätzlicher Bildschirm an der dem Sofa – also dem Sitzplatz der Versuchsperson – gegenüberliegenden Wand aufgehängt.

Bild 3.1 RUB-IKA Testsite, Versuchsraum

Bild 3.2 Versuchsraum, Ansicht aus der Gegenrichtung

Vor Beginn des Experiments wurde den Versuchspersonen erklärt, was man mit dem INSPIRE-System machen kann und welcher Zweck mit dem Versuch verbunden sei. Dann wurden Erfahrungen, die die Versuchsperson bisher mit Sprachtechnologien, also Spracherkennern, Sprachsynthese und Sprachdialogsystemen, gemacht hat, anhand eines Fragebogens erfasst. Mittels einer Kurzgeschichte, in der die Benutzung des Systems beschrieben wird, sollte der Teilnehmer nun ein „Mind Setting“ aufbauen, sich in die Problematik des Smart-Home-Systems hineindenken.

Jeder der 24 Teilnehmer führte 3 Dialoge (Interaktionen) mit INSPIRE, wobei jeweils alle drei Metaphern abgedeckt wurden. Für die Interaktionen waren drei verschiedene Szenarios definiert, die eine Situation beschreiben, in der der Benutzer eine Folge von Aktionen mit dem System durchgeführt. Die Beschreibung der Szenarien erfolgte im Stil „Sie kommen nach Hause, Ihnen ist warm. Benutzen Sie den Ventilator zur Kühlung.“ Die Versuchspersonen sollten also in eine plausible Situation für den Gebrauch des Systems versetzt und mit der Benutzung, dem Zweck und der Funktionalität des Systems vertraut gemacht werden. Bei der *Formulierung* der Aufgaben wurde darauf geachtet, dass der Wortlaut möglichst kein bestimmtes Vokabular bzw. Kommando suggeriert. In einigen Fällen wurde dem Benutzer eine geringe Entscheidungsfreiheit bei der Durchführung der Aufgaben. So lautet z. B. eine Aufgabe: „Wählen sie einen Film aus, der Ihnen gefallen könnte.“ Die vollständigen Szenarios sind im Appendix (7.1) wiedergegeben.

Jedes Szenario enthält 11-12 Aufgaben, von denen einige mehr, andere weniger komplex sind. Zu den weniger komplexen Aufgaben gehört das Ein- und Ausschalten des Ventilators und des Fernsehers, sowie die Bedienung des Rollos, das hoch und runter gefahren werden und durch das Kommando „Stopp“ in einer beliebigen Position angehalten werden kann. Bei diesen Aufgaben traten nur wenige Probleme in der Interaktion auf.

Zu den komplexeren Aufgaben gehörte zum einen die Benutzung des EPG. Mit dieser, in der Szenariobeschreibung „Programminformation“ genannten Funktion des Fernsehers kann der Benutzer eine Fernsehsendung finden, indem er den Sendetag, die Sendezeit und die Programmart spezifiziert. Die nach Festlegung dieser Suchkriterien verbleibenden

Sendungen werden in einer nummerierten Liste auf dem Bildschirm ausgegeben, aus welcher der Benutzer die präferierte Sendung wählen kann, indem er die entsprechende Zahl nennt. An diesem Punkt des Dialogs versteht das System nur Nummern, d. h., es kann kein anderes Attribut eingegeben werden. In einem weiteren GDN muss der Benutzer dann die „Programmhandhabung“ wählen, wobei die Möglichkeiten bestehen, die Sendung anzuschauen, aufzunehmen oder an den Beginn zu erinnern. Auch hier deutet sich schon in der Beschreibung der Funktionsweise und der Art der Formulierung der Aufgabe an, dass diese den Versuchspersonen häufig Schwierigkeiten bereitete. Insbesondere erwarteten viele, bei der Benutzung des EPG mit Hilfe des Namens einer Sendung Informationen zu dieser (z. B. Startzeit) zu finden und reagierten verunsichert, wenn das System diese zuerst abfragte.

Weiterhin bereitete den meisten Testteilnehmern die Bedienung der Lampen Probleme, wobei in diesem Fall die Schwierigkeit darin bestand, dass die drei Leuchten rein verbal und für das System verständlich beschrieben werden mussten. Da es 2 Stehlampen (also Lampen gleichen Typs) gab, musste zusätzlich zum Gerätenamen eine in der Beschreibung kompliziertere Eigenschaft wie Standort oder Farbe zur Identifizierung herangezogen werden (Kommandos wie „eine der Stehlampen“ sind selbstverständlich nicht möglich, da das System keine Entscheidung treffen kann). Weiter befanden sich zwei der Lampen rechts von der Versuchsperson und zwei hatten die gleiche Farbe, so dass außer bei der Tischlampe immer zwei Eigenschaften gleichzeitig bestimmt werden mussten. Zusätzlich wurden die Lampen-Aufgaben dadurch erschwert, dass in allen drei Szenarios genau zwei der drei vorhandenen Lampen benutzt werden mussten. INSPIRE kann jedoch nur jeweils eine oder alle drei Lampen zusammen steuern. Nicht zuletzt weil die Szenariobeschreibungen die Lampen als Paar behandeln (z. B. „Sie sorgen mit 2 Lampen für eine gute Beleuchtung, aber reduzieren deren Helligkeit.“), bestand eine weitere Schwierigkeit darin, zu erkennen, dass die beiden Lampen nur einzeln, also nacheinander bedient werden können.

Die Aufgaben mit dem Anrufbeantworter waren prinzipiell nicht so komplex wie die mit den Lampen und dem EPG, aber das Vokabular des Systems für den Anrufbeantworter war sehr eingeschränkt. So musste zum Beispiel in der Antwort auf die Frage „Ich habe

Anrufbeantworter als Gerät verstanden. Was kann ich für Sie tun?“ nochmals das Gerät und dann die Aktion genannt werden (z. B. „Anrufbeantworter abfragen.“). Für das Speichern bzw. löschen, sowie das Springen zur nächsten Nachricht, mussten festgelegte Kommandos benutzt werden, die dem Benutzer nach Abspielen der Nachricht angeboten wurden. Damit kommt dem Anrufbeantworter gewissermaßen eine Sonderrolle zu, da das INSPIRE System ansonsten freie Äußerungen des Benutzers versteht.

Nach jedem der drei Szenarios füllten die Versuchspersonen einen Fragebogen zu der gerade beendeten Interaktion aus. Am Ende des gesamten Experiments wurde das INSPIRE-System von der Versuchsperson mit einem weiteren Fragebogen bewertet.

3.1.3 Benutzerurteile

In dem Fragebogen „Beurteilung der Interaktion“, der jeweils nach den Dialogen ausgefüllt wurde, wurden insgesamt 37 Benutzerurteile abgefragt. Das erste Urteil betrifft den Gesamteindruck der Interaktion, die weiteren Bewertungen sind in 6 Gruppen gegliedert:

- Erreichen der gewünschten Ziele
- Verständigung mit dem System
- Verhalten des Systems
- Gespräch
- Persönliche Wirkung
- Benutzbarkeit des Systems

Die Gliederung der Fragen auf dem Fragebogen entspricht nicht den tatsächlichen Bewertungs- bzw. Wahrnehmungsdimensionen der Probanden. Diese wurden später mittels einer Faktorenanalyse im Rahmen der Auswertung des Experiments für das INSPIRE-Projekt ermittelt (Boland et al. 2006) und beinhalten

- Akzeptanz
- kognitive Anforderung
- Systemfehler
- Einfachheit der Bedienung

- eine Komponente, die mit den Aufgaben zu tun hat
- Systemverhalten
- Schnelligkeit des Systems und
- eine Komponente, die nicht erklärt werden konnte.

Die Ergebnisse der Faktorenanalyse werden in Kapitel 3.3.2 ausführlich wiedergegeben.

Die Fragen selbst waren als Statements formuliert, um von Versuchspersonen, die tendenziell unentschieden urteilen, ein deutlicheres Votum zu provozieren (vgl. Möller 2005b). Zur Beantwortung der Fragen standen fünfstufige, diskrete Skalen zur Verfügung, wobei die Skalenstufen mit den Begriffen „stimme stark zu“, „stimme zu“, „unentschieden“, „lehne ab“ und „lehne stark ab“ versehen waren. Lediglich die Frage nach dem Gesamteindruck der Interaktion wurde auf einer kontinuierlichen, jedoch auch fünfstufigen Skala beantwortet, die mit den Begriffen „schlecht“, „dürftig“, „ordentlich“, „gut“ und „ausgezeichnet“ beschriftet war. Entsprechend war die Frage nicht als Statement formuliert.

Der Fragebogen vor Beginn des Experiments soll hier nicht näher erläutert werden, da er noch keine Urteile zum Umgang mit dem INSPIRE System beinhaltet – die ja das Ziel der Vorhersage darstellen – und deshalb für diese Arbeit nicht verwendet wurde. Für spätere Analysen, die komplexere Zusammenhänge (z. B. zwischen Erfahrung mit SDS, Fehlern und Qualitätsurteilen) untersuchen oder auf die Fehler selbst zielen (z. B., machen Versuchspersonen mit mehr Erfahrung weniger Fehler), mag er dennoch nützlich sein. Auch der Fragebogen am Ende des Experiments soll hier nicht weiter betrachtet werden, da mit diesem zum einen das System bewertet wird, während sich die Untersuchung der Fehler auf die Interaktionen bezog. Die Fragen sind also weniger passend als die bei der Interaktionsbewertung. Zum anderen liegt nur ein abschließendes Urteil gegenüber drei Interaktionen pro Proband vor. Hat also eine Versuchsperson drei sehr unterschiedliche Dialoge geführt, wird durch die gleiche Bewertung der Dialoge ein sinnvolles Ergebnis erschwert. Die Einbeziehung dieser Urteile ist deswegen eher bei einer benutzerweisen Auswertung sinnvoll.

Alle für den Test verwendeten Fragebögen wurden von der Projektgruppe speziell für diesen entworfen. Dabei bezog man sich auf frühere Erfahrungen mit dem Design von Fragebögen, z. B. aus den Tests von Möller (2005b), in denen das auch in dieser Arbeit behandelte BoRIS-System bewertet wurde. Weitere Anregungen kamen von dem sog. SASSI-Fragebogen, der von Hone und Graham (2000) für Systeme mit Spracheingabe entwickelt wurde, allerdings keine Sprachausgabe berücksichtigt. Der detaillierte Fragebogen ist im Appendix (7.2) abgedruckt.

Beim Übertrag in die Datenbank wurden den Urteilen Zahlen von -2 bis +2 zugeordnet. Dabei wurden positiven Zahlen positive Urteile zugeordnet, d.h. lautet die Frage z. B. „Das System tat nicht immer was ich wollte“ und wird beantwortet mit „stimme zu“, ist der Eintrag in der Datenbank “-1“. Eine Ausnahme bildet wiederum das Urteil zum Gesamteindruck, das mit den Zahlen von 1 bis 5 codiert wurde, wobei eine 5 mit der Bewertung „ausgezeichnet“ gleichzusetzen ist.

3.2 Fehlerklassifikation

3.2.1 Ausgangspunkt

Der Fehlerklassifikation, die hier beschrieben werden soll, und die vom Autor mitentwickelt wurde, basiert auf der Vorarbeit von Antti Oulasvirta, der diese bei den Deutsche Telekom Laboratories entwickelte. Hier soll zuerst das „Coding Manual Version 1“ beschrieben werden, das den Ausgangspunkt für die eigene Arbeit an der Fehlerklassifikation darstellte.

Zu den Grundsätzen des Dokuments gehört, dass das SDS gewissermaßen als unflexibles Objekt betrachtet wird, das vom Benutzer gebraucht werden kann, um ein bestimmtes Ziel zu erreichen. D. h., das SDS hat verschiedene Eigenschaften, die vom Benutzer so hingenommen werden müssen, weil er sie nicht verändern kann. Beim Umgang mit dem System muss sich also der Benutzer diesem Anpassen (und nicht umgekehrt). Man kann weiter davon ausgehen, dass das Ziel des Benutzers ist, sich dem System bestmöglich

verständlich zu machen[10]. Deshalb kann man letztlich eine misslungene Handlung am System einem Fehler des Benutzers beim Verständnis oder der Benutzung des Systems zuschreiben. Selbstverständlich versucht der Systementwickler das System so zu entwerfen, dass der Benutzer es möglichst problemlos benutzen kann. Ein Fehler in der Interaktion ist so gesehen immer auch ein Designfehler, eine Abweichung vom idealen System. So kann zum Beispiel die Verwendung einer falschen Vokabel (aus der Sicht auf den Benutzer) auch als das Nichtverstehen dieser (aus der Sicht auf das System) bezeichnet werden. Das im „Coding Manual Version 1“ zum Ausdruck gebrachte Klassifikationsschema folgt jedoch der erstgenannten Perspektive.

Ein weiterer Grundsatz besteht darin, dass die Benutzerfehler nach ihren Phänotypen unterschieden werden, dass also das Verhalten des Benutzers klassifiziert wird, wohingegen die für den Fehler verantwortlichen mentalen Prozesse unberücksichtigt bleiben. Dies kann damit begründet werden, dass im Gegensatz zu den Genotypen, also der Ursächlichkeit von Fehlern, die - im hier vorliegenden Fall der Beurteilung von Dialogen anhand von deren Transkriptionen - nicht bekannt sind und nur mit einer begrenzten Sicherheit aus dem Gesprächsverlauf abgeleitet werden können, die Beurteilung nach Phänotypen die Sicherheit der Beobachtbarkeit der Zugehörigkeit zu einer Klasse gewährleisten sollte[11].
Die Klassifikation der Fehler sollte zudem verschiedenen Kriterien genügen, die durch die Verwendung in späteren Projekten der Telekom Laboratories vorgegeben sind. Die Fehlerklassen sollten demnach möglichst

- systemübergreifend gültig sein,
- mit den technischen Systemmöglichkeiten oder dem Mentalen Modell des Benutzers zusammenhängen,
- Probleme beschreiben, die automatisch generiert werden können und
- einen messbaren Zusammenhang mit der Benutzbarkeit des Systems und der vom Benutzer erfahrenen Qualität haben.

[10] Eine Ausnahme bildet der Fall, dass der Benutzer mit dem System „spielt“, z. B. um es zu testen (d.h., er sucht Systemfehler)
[11] Vgl. hierzu die oben (Kapitel 2.1) gemachten Bemerkungen zum Konzeptuellen Modell des Forschers vom Mentalen Modell des Benutzers

Die letztgenannte Anforderung konnte beim Entwurf der Klassifikation nur grob abgeschätzt werden und wurde deshalb vorerst nur sekundär berücksichtigt. Eine Auswertung der Beziehungen zwischen den letztendlich gebildeten Fehlerklassen und den Qualitätsurteilen erfolgt in Kapitel 3.3. Die zweite Anforderung wurde gegenüber einer guten Klassifizierbarkeit der Fehler vernachlässigt. Eine genotypische Klassifikation hätte sicher einen engeren Bezug sowohl zu den Mentalen Modellen des Benutzers als auch zu den Systemmöglichkeiten bzw. der tatsächlichen Funktionsweise des Systems ergeben. Immerhin spiegelt sich diese Anforderung aber darin wieder, dass das Benutzerverhalten annotiert wird. Als hauptsächliche Richtlinien bei der Klassifikation bleiben schließlich diese, dass sie systemübergreifend sein soll und die Fehler automatisch generierbar sind.

Eine zuverlässige Definition dessen, was einen Fehler ausmacht, gab es zu diesem Zeitpunkt noch nicht. Ein Fehler wurde zwar als Abweichung von dem „optimalen Lösungsweg“ definiert, letzterer Begriff blieb jedoch unerklärt. Ein optimaler Lösungsweg lässt sich insbesondere wegen der Eigenschaft des Systems, dem Benutzer teilweise die Dialoginitiative zu überlassen, nicht leicht bestimmen. Deshalb wurde die Entscheidung, ob eine Äußerung als fehlerhaft zu betrachten ist, vom Annotierer intuitiv gefällt. Dabei entwickelte sich eine Praxis heraus, die wie folgt beschrieben werden kann: Zuerst wird festgestellt, welche AVPs die Intention des Benutzers bestmöglich beschreiben, wobei diese nicht unbedingt den Systemmöglichkeiten entsprechen müssen. Die Intention des Benutzers wird aus der aktuellen Aufgabe des Test-Szenarios und möglichen Bedeutungen der Äußerung kombiniert. Die vorgeschlagene AVP-Konstellation wird dann mit den tatsächlich vom System im Folgezustand hinzugewonnenen AVPs, also den für das Voranschreiten des Dialogs relevanten, verglichen. Sind diese nicht gleich, wird ein Fehler annotiert.

Die initialen Definitionen der Fehlerkategorien, die auf theoretischen (also prinzipiellen) Überlegungen beruhen, lauten wie folgt:

1. Capability
- def= Issuing a command for action that is grammatically valid but the intended action cannot be performed by the system because it does not possess that capability. It is possible to conceive an extension to the system that would be able to perform the action.
- Example: Asking the system to buy new milk to the refrigerator.
- Example: Asking the system to dim a light that actually can be controlled only at the on/off level.

2. State
- def= Issuing a command that is valid and progressive in one state of the system, but not in the current one.
- Example: Asking the system to play a message although it should be first stated that the answering machine is to be operated.

3. Vocabulary & Grammar
- def= Issuing a command that would be valid if one word was changed to its synonym or the grammatical order of words was changed.
- Example: Asking for "presenting" the message instead of "playing" it.

4. Modelling
- def= Issuing a command that would be valid if the system modelled the world in a different way.
- Example: The user asks TV programme for the evening by asking the system to show programmes after 5pm. However, the system models time by categorizing it to morning, afternoon, evening, and night, and does thus not know the meaning of 5pm although it is in practice the same than "evening".[12]

Da ein Zusammenhang zwischen den aufgetretenen Fehlern und den Benutzerurteilen möglicherweise über die jeweiligen Konsequenzen der Fehler besteht, wurden auch diese klassifiziert.

[12] Zitiert aus internem Arbeitsmaterial, erstellt von Antti Oulasvirta, HIIT Helsinki, s. auch Oulasvirta et al. 2006.

1. Stagnation. The system takes the user to a prompt that is however as close to the task goal as the previous prompt. I.e., the goal can still be reached with as many steps as before.
- A special case of this is called Repetition. The system repeats the same prompt.

2. Regression. The system goes to a state that is farther away from the task goal than the previous state. I.e., the user is deviated from an optimal solution path and now has to go through at least one extra state in order to achieve to the goal.
- A special case of this is called Restart. The system returns to its initial state, losing any progress achieved in the task before the error occurred.[13]

Für die Annotation des Dialogmaterials aus dem Experiment mit dem INSPIRE-System wurde eine Excel-Datei erstellt, die wie folgt beschrieben werden kann. Ein erster Block enthält pro Zeile einen System-Prompt und die darauf folgende Benutzeräußerung. Dazu kommen Informationen, die den aktuellen Wortwechsel (also die Zeile im Datenblatt) eindeutig identifizieren (Benutzernummer, Szenario ID, Systemmetapher, Wechselnummer) sowie eine Spalte für das gerade verwendete Gerät und die Nummer der Aufgabe, die gerade bearbeitet wird, innerhalb des Szenarios. Der zweite Block enthält eine Spalte für jede Fehlerklasse, in der die Zugehörigkeit mit einer „1“ gekennzeichnet wird, andernfalls wird eine Null eingetragen. Schließlich ein Block, in dem die Konsequenzen der Fehler in der selben Weise eingetragen werden.

3.2.1.1 Beispiele

An einigen Beispielen soll verdeutlicht werden, wie die beschriebenen Fehler beim INSPIRE-System konkret aussehen können:

Capability

Da die Aktionen, die mit den Geräten durchgeführt werden sollten, weitestgehend in den Szenarios definiert waren, trat der CAPABILITY-Fehler nur (a) mit der rechten Stehlampe, die im Gegensatz zu den anderen beiden nicht gedimmt werden konnte, und (b) dem

[13] Ebd.

INSPIRE-System selbst auf. In letzterem Fall versuchte die VP entweder, gleichzeitig zwei Aktionen mit dem selben Gerät zu machen, oder mit zwei Geräten die gleiche Aktion durchzuführen.

S: Was kann ich noch für Sie tun?
U: Lampe rechts neben dem Sofa Helligkeit reduzieren.
S: Ich habe Lampe als Gerät und hinten rechts als Standort verstanden. Was kann ich für Sie tun?
U: Helligkeit reduzieren.
S: Die Lampe hinten rechts kann nicht gedimmt werden. Was kann ich noch für Sie tun?

S: Was kann ich noch für Sie tun?
U: Schalte bitte alle Lichter und den Fernseher wieder aus.
S: Ich habe Fernseher als Gerät, alle als Standort und ausschalten als Aufgabe verstanden. Leider kann ich diese Aufgabe nicht erfüllen. Bitte wählen Sie eine der Möglichkeiten auf dem Bildschirm aus, indem sie die zugehörige Nummer sagen.
[Auf dem Bildschirm werden mögliche Lösungen mit Teilmengen der aktuellen AVPs dargestellt]
U: Eins.
S: Was kann ich noch für Sie tun?
U: Den Fernseher noch ausschalten.
S: Was kann ich noch für Sie tun?
U: Ich hätte gerne Licht. Schalt bitte die Lampe links vom Sofa ein und dimme sie.
S: Die Lampe links ist bereits dunkel. Was kann ich noch für Sie tun?

Das INSPIRE-System kann die Lampe *entweder* dimmen *oder* einschalten, da beide Aktionen über verschiedene Werte für das selbe Attribut definiert sind. Hier wird der Wert „einschalten“ von dem später genannten Wert „dimmen“ übeschrieben, so dass das System versucht, die ausgeschaltete Lampe dunkler zu stellen.

State

Weil das System prinzipiell auch AVPs verstehen kann, die nicht mit dem im aktuellen GDN zu bestimmenden Attribut korrespondieren, kann dieser Fehler nur auftreten, wenn dieses Prinzip gebrochen wird. Dies ist der Fall bei den so genannten „List-Processing-Nodes“, in denen eine Zahl für einen Fernsehsendungstitel genannt werden soll und auch nur diese Zahl verstanden wird, und in den GDNs, in denen mit „ja“ oder „nein“ geantwortet werden kann, also den Confirmation-GDNs und den Reconsideration-GDNs.

S: Ich habe Spielfilm als Programmart verstanden. Aus Ihrer Wahl ergeben sich mehrere Möglichkeiten. Bitte nennen Sie die Nummer eines Titels aus der Liste auf dem Bildschirm.
[Auf dem Bildschirm wird eine Liste der zur Auswahl stehenden Sendungen dargestellt]
U: Die eins, und erinner' mich an den Anfang.
S: Ich habe Das dreckige Dutzend als Titel verstanden. Was möchten Sie mit der ausgewählten Sendung tun?
U: Erinner' mich dadran, wenn die Sendung anfängt.
S: Okay. Was kann ich noch für Sie tun?

Hier versucht der Benutzer in einem List-Processing-GDN, zusätzlich zu dem gewünschten Titel auch die Programmhandhabung zu spezifizieren. Das System ignoriert diesen zweiten Teil der Äußerung jedoch.

Vocabulary & Grammar

Vokabular- und Grammatikfehler tauchen in den unterschiedlichsten Formen auf, wie das folgende Beispiel illustriert:

S: Was kann ich noch für Sie tun?
U1: Hab ich Nachrichten auf dem Anrufbeantworter?
S: Ich habe Anrufbeantworter als Gerät verstanden. Was kann ich für Sie tun?
U2: Abhören bitte.
S: Was kann ich für Sie tun?
U3: Nachrichten vorlesen.
S: Was kann ich für Sie tun?
U4: Welche Nachrichten hab ich? Nachrichten abfragen.
S: Ich habe aktuelle Nachricht als Aufgabe verstanden. Nennen Sie bitte Ihren Namen.

Im Fall von U1 sollte das System idealerweise verstehen, dass der Benutzer die neuen Nachrichten, falls es welche gibt, abhören möchte. Da das Nomen „Nachrichten" hier nicht zu einem AVP zugeordnet wird, um Verwechslungen mit den Fernsehnachrichten zu vermeiden, versteht das System hier nur das AVP {Gerät= AB}.
Die folgende Äußerung U2 wird vom System gar nicht verstanden. Das Verb „abhören" kann zwar prinzipiell verwendet werden, aber nur, wenn es auf „Anrufbeantworter" oder „Nachrichten" folgt.

Auch U3 wird nicht verstanden, da in diesem Fall das Verb „vorlesen“ nicht erkannt wird. Erst in der folgenden U4 verwendet der Proband eine gültige Kombination: „Nachrichten abfragen“.

Modelling

Modellierungsfehler kommen in dem untersuchten Datensatz bei Referenzen zu Raum oder Zeit vor. Im Prinzip, d.h. hinsichtlich der Funktionsweise des Systems, sind diese Fehler den Vokabularfehlern sehr ähnlich, da die Modelle von Zeit und Raum, die das System kennt, vollständig durch Einträge in der MappingNLU-Datei konstituiert werden. Der Unterschied zu Vokabularfehlern besteht jedoch darin, dass die Einträge für Raum- und Zeitinformation von einer relativ offensichtlichen entsprechenden Konzeption des Entwicklers geleitet werden. Dies bedeutet, dass ein Ersetzen des Vokabulars durch Synonyme nicht zur Vermeidung des Fehlers beiträgt. Im folgenden Beispiel beschreibt ein Benutzer den Standort einer Stehlampe als „auf dem Boden“, während der Entwickler den Raum mittels der Begriffe „vorne“, „hinten“, „rechts“ und „links“ einteilt und als Referenzobjekte das Sofa und den Tisch benutzt.

> *S: Ich habe Lampe als Gerät und ausschalten als Aufgabe verstanden. Wo steht die Lampe, die sie steuern möchten?*
> *U: Rechts auf dem Boden.*
> *S: Mögliche Positionen sind: links neben dem Sofa, rechts neben dem Sofa, oder rechts auf dem Tisch.*
> *U: Rechts neben dem Sofa aus.*
> *S: Was kann ich noch für Sie tun?*

Im Gegensatz dazu liegt kein Modellierungs-, sondern ein Vokabularfehler vor, wenn, wie im folgenden Dialog, bei Anwendung einer kompatiblen Raumkonzeption zum Beispiel das Sofa als Couch bezeichnet wird.

> *S: Was kann ich noch für Sie tun?*
> *U: Licht rechts neben der Couch an.*
> *S: Ich habe Lampe als Gerät verstanden. Welche Lampe möchten Sie steuern?*
> *U: Rechts neben Couch.*
> *S: Ich konnte Sie nicht verstehen. Sie können die Tischlampe, die weiße Stehlampe oder die gelbe Stehlampe bedienen.*
> *U: Rechts neben dem Sofa.*
> *S: Ich habe hinten rechts als Standort verstanden. Was kann ich für Sie tun?*

3.2.2 Anpassung des Klassifikationsschemas

Nicht alle Äußerungen, die als fehlerhaft identifiziert wurden, konnten ebenso leicht und eindeutig in eine Fehlerkategorie eingeordnet werden wie die oben dargestellten Beispiele. Die beschriebene Definition der Fehler und Konsequenzen wurde deshalb in mehreren Schritten an die Probleme und Unstimmigkeiten, die bei der Annotation auftraten, angepasst. So wurde zuerst eine neue Fehlerklasse eingeführt, um die Wechsel zu beschreiben, in denen der Benutzer nicht antwortet (NO INPUT). Des Weiteren wurde eine Klasse für nicht klassifizierbare Fehler nötig (OTHER). Zudem wurden alle bisherigen Fehlerkategorien in mehrere Unterkategorien gespalten, wobei die Bedeutung der Kategorien jedoch erweitert wurde. Dadurch ließen sich letztendlich auch bisher kritische Fälle definitionsgemäß einordnen.

So wurde zum Beispiel die Klasse CAPABILITY in die Unterklassen INEXISTENT OBJECT OF CONTROL, INADEQUATE MAGNITUDE OF CONTROL und CONTROL CONFLICT unterteilt. Während die ursprüngliche Definition im Prinzip den INADEQUATE-MAGNITUDE-OF-CONTROL-Fehler und den INEXISTENT-OBJECT-OF-CONTROL-Fehler umfasst, wurde ein Kommando, das vom System eine prinzipiell unmögliche Aktion fordert (z. B. einen Film zeigen, der gerade nicht ausgestrahlt wird) nicht eindeutig durch diese erfasst (jetzt CONTROL-CONFLICT-Fehler). Die Fehlerklasse INEXISTENT-OBJECT-OF-CONTROL beschreibt nun den Fall, dass der Benutzer ein Gerät zu benutzen versucht, dass nicht mit dem System gesteuert werden kann, während bei INADEQUATE-MAGNITUDE-OF-CONTROL der Benutzer ein vorhandenes Gerät in einer Weise zu steuern versucht, die die Möglichkeiten des Geräts übersteigt. Als Gemeinsamkeit dieser drei Fehler bleibt jedoch, dass das Ziel des Benutzers die Möglichkeiten des Systems übersteigt.

Auch der STATE-Fehler wurde um eine weitere Klassendefinition angereichert. Als „UNPROGRESSIVE STATE ERROR“ wurde von nun an eine Äußerung bezeichnet, die zwar gültig, in dem betreffenden Zustand des Systems jedoch nicht progressiv ist, da sie keine für das System neuen Information enthält. Das Kriterium der Progressivität war für den „normalen“ STATE-Fehler angenommen worden, um den Fall aus der Definition auszuschließen, dass der Benutzer ein Kommando gibt, dass zwar gültig ist, aber nicht geeignet, um die aktuelle Aufgabe zu erfüllen.

Beim MODELLING-Fehler wurde die Definition zwar nicht erweitert, es wurden jedoch etwas konkretere Typen von Fehlern spezifiziert, die nach der Art der modellierten Domäne unterscheiden. Dadurch wurde die Definition der MODELLING-Kategorie etwas klarer, die in der ersten Fassung zu Missverständnissen führte, da hier keine Modelle des Systems selbst oder der Repräsentation der Aufgaben gemeint waren. Diese sollten deshalb ausgeschlossen werden, weil sie kein eindeutiges phänotypisches Erscheinungsbild haben. Damit unterläge die Annotation zu einem gewissen Grad der Spekulation. Später wurde jedoch mit dem ATTRIBUTE-TYPE-Fehler doch ein Teil der Aufgabenmodellierung in die MODELLING-Klasse aufgenommen. Dieser wird annotiert, wenn der Benutzer die Domäne anders kategorisiert als das System (z. B. Actionfilm als Programmart wählt, während das System nur eine Kategorie „Spielfilm“ benutzt) oder eine falsche Variablenklasse benutzt (z. B. Filmtitel anstatt der zugehörigen Zahl).

Die VOCABULARY-AND-GRAMMAR-Kategorie wurde vor allem deshalb in Unterklassen geteilt, weil auf diese ein Großteil der Fehler fiel, so dass eine genauere Erfassung dieser Fehler lohnenswert erschien und als möglich betrachtet wurde. Dazu wurde zunächst zwischen Wortfehlern, Phrasenfehlern und Grammatikfehlern unterschieden. Die Wortfehler ihrerseits wurden nach der Wortart (VERB, Nomen und ADJECTIVE/ADVERBIALE BESTIMMUNG) unterteilt. Als Phrasenfehler werden Äußerungen bezeichnet, bei denen nicht einfach Vokabeln gegen Synonyme ausgetauscht werden können, um sie zu korrigieren (z. B. „Ventilator lauf los“). Grammatikfehler wurden annotiert, wenn das System die Vokabeln zwar kennt, jedoch nicht in der vorliegenden grammatikalischen Variante (z. B. versteht es „Anrufbeantworter abftragen“, aber nicht „frag den Anrufbeantworter ab“). Für den Fall, dass ein Fehler den Vokabelfehlern zugeordnet werden kann, jedoch in keine der genannten Klassen passt, wurde zusätzlich eine OTHER-Klasse geschaffen.

Auch bei der Klassifizierung der unmittelbaren Konsequenzen traten Annotationsprobleme auf. So gab es zum einen keine Möglichkeit, ein partielles Voranschreiten des Dialogs zu vermerken. Damit ist gemeint, dass bei einer Äußerung, mit der der Benutzer mehrere AVPs festlegt, nur ein Teil dieser verstanden wird. So

kommt es z. B. vor, dass das System von der Äußerung „Lampe an.“ Nur das Gerät „Lampe“ versteht, nicht jedoch die Aktion „einschalten“. Zum anderen wurde als weiterer Spezialfall von STAGNATION das Abspielen eines Hilfe-Prompts als eigene Konsequenzklasse definiert, da diese insofern einen Sonderfall darstellt, als hier das System implizit signalisiert, dass es das Problem bemerkt hat und mögliche Äußerungen mehr oder weniger konkret vorgibt.

3.2.2.1 Sonstige Fehler

Nachdem der gesamte Datenkorpus annotiert war, verblieb ein relativ großer Teil (ca. 10%) der Fehler, der nicht in eine der Klassen eingeordnet werden konnte. Ein Großteil der Fehler dieser OTHER-Kategorie konnte jedoch in drei Untergruppen eingeteilt werden. So bestand fast die Hälfte darin, dass ein Objekt (z. B. ein Gerät oder ein Film) spezifiziert wurde, indem auf einen früheren Systemstatus Bezug genommen wurde. Als Beispiele mögen die Äußerungen „schalte die andere Lampe auch ein“ oder „zeichne den Film auf, an den du mich erinnern solltest“ dienen. Dieser Fehler wurde vorerst als „REFERENCE“-Fehler bezeichnet. Von den übrigen Fehlern bestanden ca. zwei Drittel darin, dass der Benutzer unbeabsichtigt eine Antwort gab, die ihn von der Lösung der aktuellen Aufgabe entfernte. Ein einfaches Beispiel dafür wäre, dass er eine korrekte Lösung im Confirmation-GDN widerruft. Etwas komplizierter zu identifizieren war dieser Fehler, wenn er in Reconsideration-GDNs auftauchte. Hier bedeutet eine „regressive Antwort“, dass es der Benutzer ablehnt, ein falsch spezifiziertes Attribut zu ändern. Schließlich konnten noch einige Fehler auf den Wizard-of-Oz“ zurückgeführt werden, so zum Beispiel Tippfehler bei der Eingabe der Benutzeräußerung.

3.2.2.2 Klassifikationsschema für die Annotation der INSPIRE-Dialoge

Aus den o. g. Veränderungen an dem anfänglichen Klassifikationsschema resultierte folgende Fassung, die für die Annotation der Dialoge mit dem INSPIRE-System verwendet wurde:

DEFINITIONS OF ERRORS

No Input
def= Failing to issue a command during the response interval where the system expects it to be issued.

1. Capability
def= Issuing a command for action that cannot be performed by the system because it does not possess that capability. It is possible to think of an extension to the system that would be able to perform the intended action.Three kinds of capability errors are distinguished:

Inexistent Object of Control Error
Attempting to control an object (device or content) that does not exist in the system. Example: Asking the system to buy new milk to the refrigerator. (One can easily think that there was a module in an improved version of the system that could do this.)

Inadequate Magnitude of Control Error
Attempting to control at a level not possible for the system. Example: Asking the system to *dim* a light that actually can be controlled only at the *on/off* level. (One can easily think that an improved system could do this.)

Control Conflict Error
Asking the system to do something which it cannot because of external restrictions. Example: Asking the system to show a film broadcasted in the evening *now*, the presentation time of which however is not controlled by the system but some external agent (TV company). (One can think that an improved system could retrieve the film on demand.)

2. State
def= Issuing a command that is valid and progressive (in regard with the goal expressed in the task given to the user in the experiment) in one state of the dialogue, but *not* in the current one. The progressiveness criterion can be compromised in some cases. It should be marked as Unprogressive State Error then.

State Error
Example: Asking the system to play a message although it should be first stated that the answering machine is to be operated.

Unprogressive State Error (a special case of the State Error)
Here the progressiveness criterion is loosened, i.e. the command has to be valid, but the corresponding AVPs have been acquired already. Example:
S: „Ich habe ANY als Programmart verstanden. Aus Ihrer Wahl ergeben sich mehrere Möglichkeiten. Bitte nennen Sie die Nummer eines Titels aus der Liste auf dem Bildschirm.“
U: “Programminformation.”

3. Vocabulary &Grammar

def= Issuing a command that would be valid if one word was changed to its synonym or the grammatical order of words was changed without changing the vocabulary nor the meaning of the utterance. Three kinds of Vocabulary & Grammar errors are distinguished:

Word Error

Issuing a command that would be valid if a word was changed to a synonym or expression with same meaning. Example: Asking for "presenting" the message instead of "playing" it.

-This type of error again can be divided into

- verb
- noun
- adjective/"adverbiale Bestimmung"

Phrase Error

. …if a phrase was changed to a phrase or word with practically the same meaning, but (partly) different vocabulary. In this case, it is not possible to exchange single words to get a valid utterance with the same meaning. Example: "Hab ich Nachrichten auf dem Anrufbeantworter"

(Includes also colloquial expressions like "…liebes Smart Home System INSPIRE")

Syntax Error

. …if the order of words or the grammatical construction was changed without changing the meaning of the totality. (The vocabulary of the utterance is known by the system.) Example: "Frag den Anrufbeantworter ab!" (should be „Anrufbeantworter abfragen.")

(Other)

4. Modelling

def= Issuing a command that would be valid if the system represented the world in a different way. It is possible to imagine another kind of model/categorization of the world, this error would not emerge. (This should not be confused with (1) the state error, wherein order errors are related to dialogue structure, not the world and (2) vocabulary errors, wherein errors cannot be drawn back to modelling of the world.). Three kinds of Modelling errors are distinguished:

Temporal Categorization Error

The user refers to categorization of events in time that is not understandable by the system. Example: The user asks TV programme for the evening by asking the system to show programmes after 5pm. However, the system models time by categorizing it to morning, afternoon, evening, and night, and does thus not know the meaning of "after 5pm" although it is in practice the same as "evening".

Spatial Categorization Error

The user refers to the space in a way that is not understandable by the system. Example: "Please turn on the lamp that is on the right from the table lamp" (This fails because the system does not have a model of the relative positions of the lamps.) (Note: This should not be confused with vocabulary error in the case the reference is made in just one word.) Example: "Please turn on the lamps on the right" – The system knows that there are two lamps on the right, but does not model their (common) relationship from the perspective of the user.

Attribute Type Error

The user categorizes the value domain of a particular attribute (GDN) in a way that is not understandable by the system. Example: specifying comedy as TV show type, uttering TV show name instead of number.

Other
All other obvious errors. Explanation for classifying this as an error should be given. E.g., system misinterpretation.) There are three cases, where an explanation is provided:

Reference
The user tries to specify a value by relating it to an earlier system state, which the system does not remember (e.g. “switch on the other lamp”, “please record the film I asked you to remind me of”).

Wrong Reply
Issuing a command that is valid in the current state of the dialogue, but regressive (with regard to the goal expressed in the task given to the user in the experiment). E.g., S: ”I understood …(list of correct values)… Do you confirm that?” – U: “No”

System/Wizard error
An error has been made by the wizard (e.g. typing error) or there was a technical problem (E.g. first prompt is played twice)

IMMEDIATE CONSEQUENCES

The analysis of immediate consequences concerns the system response. The immediate effect of an erroneous utterance can be of three types:

1. Stagnation
The system takes the user to a prompt that is however as close to the task goal as the previous prompt. I.e., the goal can still be reached with as many steps as before.
- A special case of this is called ***Repetition/Rephrasing***. The system repeats the same prompt (word to word or just the end part of it but meaning the same thing and being pragmatically the same prompt with same action alternatives (E.g., "Was kann ich fuer Sie tun" is an often repeated shorthand for more complex prompts.).
- Another special case of this is called ***Help***. The system plays the help prompt of the current GDN (in which the action alternatives are proposed to the user)

2. Regression
The system goes to a state that is farther away from the task goal than the previous state. I.e., the user is deviated from an optimal solution path and now has to go through at least one extra state in order to achieve to the goal.
- A special case of this is called ***Restart***. The system returns to its initial state, losing any progress achieved in the task before the error occurred. (*latent* consequences will be automatically analysed based on the coded transcripts.)

3. Partial Progress
The system goes to a state which is closer to the task goal, but not all the information in the user's utterance is processed.[14]

3.2.2.3 Verbleibende Probleme bei der Zuordnung der Fehler zu Klassen

Trotz der sorgfältigen Anpassung an die Probleme, die bei frühen Kodierungsdurchgängen auftauchten, blieben einige Schwierigkeiten bei der Annotation. Zu einem gewissen Teil beruhen diese wohl auf den Prinzipien der Klassifikation selbst, nämlich dass sie phänotypisch und aus der Sicht auf den Benutzer kategorisiert. Dies ist in vielen Fällen wenig intuitiv, da der Benutzer im Prinzip den Maßstab für die Kommunikationsfähigkeit darstellt, an dem sich das System messen muss. Die Intuition trübt dann wiederum den definitionsgemäßen Blick auf das Problem. Weiterhin sind die Fehlerklassen teilweise überdefiniert, d. h., um alle Fehler auszuschließen, die nicht in die Klasse hineingehören

[14] Zitiert aus internem Arbeitsmaterial, Antti Oulasvirta und Klaus-P. Engelbrecht, Deutsche Telekom Laboratories, 2006

sollen, sowie um einander ausschließende Klassen zu haben, wurden die Definitionen mehrmals verfeinert und stellten sich im Ergebnis dieses Prozesses als etwas konstruiert heraus. Z. B. ist es kein STATE ERROR, im Start-GDN nach der „Heute“-Sendung zu fragen, weil das System den Sendungsnamen auch in keinem anderen Zustand versteht. Andererseits kann aber der Sendungsname in keiner Form (also auch nicht als Nummer kodiert) in einem anderen als dem zugehörigen GDN verstanden werden, so dass sich in diesem Fehler auch das Problem, das eigentlich mit der STATE-ERROR-Klasse erfasst werden sollte, manifestiert.

Der Vorteil von genauen und sich gegenseitig ausschließenden Definitionen ist jedoch, dass die Reliabilität der Annotation leichter auf einem hohen Niveau gehalten werden kann. Nimmt man die Definitionen nicht wörtlich, können viele der Fehler nicht eindeutig in eine Klasse eingeordnet werden, wie das oben angeführte Beispiel („Heute“-Sendung) zeigt. Verzichtet man aber auf die Eindeutigkeit, wird die Aufgabe des Annotierens ungleich komplexer, da jede Äußerung genauestens auf die verschiedenen Fehlerarten geprüft werden muss, wobei die Systemantwort immer nur darauf schließen lässt, dass *irgendeine* Kombination von Fehlern passiert sein muss. Man kann also leicht Aspekte der Fehler übersehen und arbeitet dadurch tendenziell ungenauer. Zudem erleichtert es die Annotation, wenn man bei Fehlern, die schwer mit der Klassifikation erfasst werden können, sich nicht die entsprechende(n) Klasse(n) merken muss, sondern diese zuverlässig und eindeutig anhand der Definitionen bestimmen kann.

Ein schwer zu klassifizierender Fehler, der innerhalb der kodierten Dialoge mehrmals auftauchte, war zum Beispiel der folgende: Die Versuchsperson antwortet auf die Frage „Was soll ich mit dem AB tun?“ mit „Abfragen.“ Das System kennt zwar im Prinzip das Verb (die richtige Äußerung wäre „Anrufbeantworter abfragen“), versteht dieses hier aber nicht, weil es nicht mit dem Begriff „Anrufbeantworter“ zusammen auftaucht. Hier lässt sich intuitiv keine Klasse eindeutig zuordnen, andererseits könnte man je nach Laune behaupten, es handele sich um einen

- Verbfehler, weil das Verb nicht verstanden wurde
- Nomenfehler, weil der Benutzer das notwendige Nomen nicht gesagt hat
- Phrasenfehler, weil eine andere Phrase verstanden worden wäre oder

- Einen sonstigen Vokabularfehler, da es sich weder um ein Nomen noch eine Phrase handelt und das Verb dem System bekannt ist

Es besteht also die Gefahr, dass der selbe Fehler in jeder seiner Instanzen anders klassifiziert wird bzw. dass sich der annotierende Experte eine Vielzahl von Sonderfällen und deren jeweilige Behandlung merken muss. Nach der Definition lässt sich dieser Fehler jedoch eindeutig als Verbfehler bestimmen.

Da offensichtlich nicht beide Anforderungen an das Klassifikationsschema, also verlässlich zu sein und die Eigenschaften der Fehler vollständig zu erfassen, gleichzeitig zu haben sind, wurde hier eine Entscheidung zu Gunsten der Konsistenz getroffen.

Ein weiteres Problem der Klassifizierung von Fehlern nach dem vorliegenden Schema besteht darin, dass der Experte sehr genau wissen muss, wie das SDS funktioniert, welche Wörter und Phrasen es versteht und auf welche AVPs diese gemappt werden, da aus dem System-Prompt, der auf die Äußerung folgt, oft nicht ersichtlich ist, worin der Fehler des Benutzers bestand. So muss man z. B., um einen STATE-Fehler zu verzeichnen, wissen, dass die Äußerung in irgendeinem anderen Systemzustand gültig gewesen wäre, während der System-Prompt nur zum Ausdruck bringt, dass sie im tatsächlich vorliegenden Zustand des Systems nicht verstanden wurde. Beim INSPIRE-System ist jedoch insbesondere die MappingNLU-Datei sehr komplex, da das Mapping ausschließlich auf Beispielen basiert und keinen konsistenten Regeln folgt, d. h., man müsste im Prinzip alle Einträge kennen oder zumindest suchen können, um zuverlässig sagen zu können, ob das System eine Äußerung verstehen wird oder nicht. Weiterhin sollte der Experte mit den Möglichkeiten des Systems vertraut sein, um z. B. über die Zuordnung zur CAPABILITY-Klasse zu entscheiden. Diese Punkte bereiteten bei der Annotierung der INSPIRE-Daten Probleme, weil die vorliegende Systemversion wesentlich ausgereifter war als die im Free-Wizard-of-Oz-Experiment getestete.

Trotz der genau definitionsgemäßen Vorgehensweise stellt sich schließlich noch das Problem, dass die Bestimmung der Klasse eines Fehlers teilweise nicht von der subjektiven Einschätzung des Experten getrennt werden kann. So könnte man z. B. fragen, ob in den Äußerungen „Ventilator lauf los“ und „Ventilator einschalten“ die Verben als

Synonyme zu betrachten sind (die Antwort entscheidet über die Zugehörigkeit zu der Verb- oder Phrasenfehler-Klasse). Oder: Ist es ein Fehler, wenn der Benutzer nach Hilfe zu den Lampen fragt, das System auch tatsächlich einen Hilfeprompt abspielt, sich aber in einem GDN befindet, dessen Hilfe-Prompt etwas anderes als die Lampenfunktionen erklärt[15]. Kann man im Fall einer positiven Antwort diesen als STATE-Fehler bezeichnen?

Von den Konsequenzen bereitete bei der Annotation vor allem die REGRESSION-Klasse Probleme, da diese über die „Entfernung" des aktuellen Systemzustands, im Vergleich zum folgenden, vom Aufgabenziel definiert ist. Kritische Fälle sind hier z. B., wenn das System ein falsches AVP extrahiert, der Dialog aber zum nächsten GDN voranschreitet. Dieses Problem wird noch komplexer, wenn zusätzlich zu dem falschen AVP auch korrekte AVPs extrahiert wurden. Da mit dem INSPIRE-System auch relativ komplizierte kommunikative Aktionen durchgeführt werden können – z. B. könnte der Benutzer das falsche Attribut neu spezifizieren – gibt es eine Vielzahl möglicher Wege zum Ziel, so dass eine konkrete Entfernung von diesem nicht bestimmbar ist. Ähnliches gilt für die Konsequenzen von Fehlern in Metakommunikations-GDNs. Hier gestaltet sich die Annotation häufig als etwas undurchsichtige Rechenaufgabe.

Schließlich gab es noch einige Äußerungen, bei denen nicht ganz klar war, ob sie als Fehler zu betrachten sind oder nicht. Im folgenden Dialog wurde z. B. ein UNPROGRESSIVE-STATE-Fehler diagnostiziert, obwohl der Benutzer sich wahrscheinlich dessen bewusst ist, dass das System „heute" bereits verstanden hat:

> *S: An welchem Tag wird die Sendung gezeigt?*
> *U: Heute.*
> *S: Ich habe heute als Tag verstanden. Um welche Uhrzeit wird die Sendung gezeigt?*
> *U: Heute abend.*

Bei der ad-hoc Fehlerdefinition, die bei der Kodierung verwendet wurde, bleibt das Problem zu erwähnen, dass die Beurteilung, welche AVPs extrahiert werden müssten,

[15] Dieses Vorkommnis ließe sich leichter annotieren, wenn man mit dem Fokus auf das System, d.h., Systemfehler statt Benutzerfehler, beurteilt. Dann handelt es sich eindeutig um einen nicht angemessenen Prompt. Dieser Aspekt der Interaktion wurde bereits im Rahmen des INSPIRE-Projektes annotiert und damit einer Analyse zugänglich gemacht. Deshalb beschränkt sich die hier beschriebene Arbeit auf Benutzerfehler.

subjektiv bleibt. Mann könnte z. B. fragen, ob die Äußerung „Abendprogramm bitte“ bedeutet, dass das Abendprogramm des heutigen oder irgendeines anderen Tages durchsucht werden soll.

3.3 Korrelationsberechnungen

3.3.1 Vorbereitung der Daten

Eine entscheidende Motivation für die Beobachtung und Klassifizierung von Interaktionsfehlern mit SDS stellt die Vorhersage von Benutzerurteilen dar. Bisher wurden verschiedene Versuche unternommen, anhand von Interaktionsparametern wie Dialogdauer auf die wahrgenommene Qualität zu schließen[16], wobei die Interaktionsparameter entweder aus automatisch generierten Log-Dateien oder aus Expertenannotationen berechnet wurden. Für das INSPIRE System liegen ca. 50 verschiedene Interaktionsparameter[17] vor, die jedoch nur eine wenig bessere Vorhersagegenauigkeit aufweisen als eine zufällige Vorhersage. Von der Einbeziehung von Fehlerzahlen und Fehlerkonsequenzen verspricht man sich eine höhere Genauigkeit der Prädiktion, zumal Interaktionsfehlerzahlen wahrscheinlich direkt mit der Qualität des Dialogs zusammenhängen (während quantitative Parameter wie Dialogdauer eher indirekt auf die Gesprächsqualität schließen lassen; so könnte man z. B. aus der Dialogdauer auf die Präsenz von Problemen, die wiederum die Qualität beeinflussen, zu schließen versuchen).

Die Voraussetzung für die Verwendung jedwelcher Parameter, also auch der Fehlerzahlen, zur Vorhersage des Urteils ist, dass die Varianz der Urteile durch die Varianz der Fehlerhäufigkeiten erklärt werden kann, oder mit anderen Worten, dass die Fehlerhäufigkeiten mit den Benutzerurteilen korrelieren. Deshalb wurden zur Überprüfung der Tauglichkeit der Fehlerklassen für die Urteilsvorhersage Korrelationen zwischen diesen berechnet. Dazu wurde die Software SPSS verwendet.

---

[16] vgl. z. B. Möller (2005b)
[17] Diese sind teilweise abhängig voneinander, z. B. „Anzahl der Benutzeräußerungen“ und „Anzahl der System-Prompts“

In SPSS stehen drei verschiedene Korrelationskoeffizienten zur Auswahl: der Pearson-Koeffizient *r*, Kendalls τ-b sowie Spearmans ρ. Der Koeffizient nach Pearson ist das gängigste dieser drei Korrelationsmaße, jedoch setzt er für beide Variablen, zwischen denen die Korrelation berechnet werden soll, Intervallskalenniveau sowie Normalverteilung voraus. Weiterhin liefert er ein zuverlässiges Ergebnis nur für lineare Zusammenhänge, d. h., nichtlineare Korrelationen werden von dem Algorithmus nicht detektiert. Der Vorteil eines viel genutzten Koeffizienten ist aber, dass die errechneten Werte mit vielen anderen Ergebnissen verglichen werden können. Kendalls τ-b erlaubt für eine der Variablen Ordinalskalaniveau oder Nicht-Normalverteilung, was im Umkehrschluss bedeutet, dass auch die Vorhersage nur eine Rangfolge bestimmen kann. τ-b ist in der Literatur äußerst selten und wird in den meisten Statistikbüchern nicht behandelt (es handelt sich offensichtlich um eine Variante des „normalen“ τ Kendalls, welches etwas gängiger ist). Aus diesem Grund wurde es für die hier beschriebene Analyse ausgeklammert. Spearmans ρ kann reliabel für zwei mindestens ordinalskalierte Variablen berechnet werden, ist also bezüglich der Datenstruktur der anspruchsloseste Koeffizient. Gleichzeitig werden auch nichtlineare Zusammenhänge detektiert, sofern sie monoton sind. Wiederum gilt, dass der Zusammenhang zwischen den Variablen auch nur auf Ordinalskalaniveau beschrieben wird. Die Werte für ρ sind gewöhnlich etwas niedriger als die für *r*.

Bei der Berechnung von Korrelationskoeffizienten spielen also die Verteilungseigenschaften sowie das Skalenniveau der Variablen eine Rolle bzgl. der Entscheidung über die geeignete Methode sowie der Auswertung der Ergebnisse. Da man gewöhnlich versucht, ein möglichst hoch skaliertes Ergebnis zu bekommen, also eine Verwendung des Pearsonschen *r* wünschenswert wäre, sollen zuerst die Verteilungseigenschaften und das Skalenniveau der Eingangsgrößen untersucht werden.

3.3.1.1 Skalenniveau und Verteilung

Während das Skalenniveau für die Fehlerzahlen leicht identifiziert werden kann – es handelt sich offensichtlich um Rationalskalen – ist dies für die Benutzerurteile etwas

schwieriger. Wie bereits erläutert, wurden die Urteile (ausgenommen r1_1 (Gesamteindruck)) auf einer fünfstufigen, diskreten Skala erhoben, deren Stufen jeweils mit einem Attribut, das den Grad der Zustimmung ausdrückt, beschriftet waren (Likert-Skala). Inwieweit die Abstände zwischen den benachbarten Attributen als gleich groß (bzw. die Skala als Intervallskala) betrachtet werden können, ist Ermessenssache und wird von verschiedenen Forschern unterschiedlich behandelt. Streng genommen ist der Abstand nicht gleich groß, weil man z. B. kaum sagen kann, dass „stimme sehr zu" eine doppelt so starke Zustimmung bedeutet wie „stimme zu". Dennoch findet man für gewöhnlich Korrelations- und andere Berechnungen auf Intervallskalenniveau mit auf Likert-Skalen erhobenen Daten. Das Skalenniveau soll deshalb nicht ausschlaggebend sein für die Verwendung eines schwächeren Koeffizienten.

Etwas kritischer sind die Verteilungsfunktionen der Variablen zu beurteilen. Die Verteilungen der Urteile wurden bereits im Rahmen der Auswertung des Free Wizard-of-Oz-Tests durch die Projektgruppe analysiert und sind in Deliverable 6.2 (Boland et al. 2004) des Projektes dargestellt. Die Daten wurden mittels des Kolmogorov-Smirnov-Tests auf Normal-, Poisson- und Exponentialverteilung untersucht. Dabei lieferten alle Variablen, die aus dem Interaktionsfragebogen gewonnen wurden ein signifikantes Ergebnis für alle drei Tests. Beim Kolmogorov-Smirnov-Test bedeutet ein signifikantes Ergebnis, dass die Variable der getesteten Verteilung nicht folgt. Eine Ausnahme bildete das Urteil über den Gesamteindruck, das auf einer kontinuierlichen Skala erhoben worden war. Da auch der K.-S.-Test mindestens intervallskalierte Daten verlangt, könnte dieses Ergebnis mit dem Skalenniveau zusammenhängen. Jedoch lieferte der Test kein signifikantes Ergebnis mehr, wenn die Fälle nach der Systemmetapher getrennt analysiert wurden.

Von den Fehlerzahlen und Konsequenzen wurden einige bereits vor der deskriptiven Analyse sofort aussortiert, da sie so selten auftraten, dass eine Korrelationsberechnung nicht sinnvoll wäre. Stattdessen wurde entschieden, für diese Klassen die Streudiagramme mit den Urteilen zu analysieren. Diese Vorgehensweise betrifft die Fehler INEXISTENT OBJECT OF CONTROL, CONTROL CONFLICT, ADJECTIVE/ADVERBIALE BESTIMMUNG, OTHER

(VOCABULARY), SYSTEM/WIZARD OF OZ und OTHER (OTHER), die alle weniger als sieben mal annotiert wurden.

Ein Kolmogorov-Smirnov-Test auf Normalverteilung ergibt für alle verbleibenden Fehler- und Konsequenzklassen außer für STAGNATION, HELP und PARTIAL PROGRESS ein signifikantes Ergebnis, d. h., außer diesen drei Klassen sind alle Fehler und Konsequenzen als nicht normalverteilt nachgewiesen. Dieses Ergebnis entspricht den Erwartungen, da die relativ selten auftretenden Fehler tendenziell einer Poissonverteilung folgen sollten. Diese Vermutung wird durch die Betrachtung der Histogramme gestützt. Deshalb wurden, dem Rat von Hair et al. (1998) folgend, vor der Berechnung von Korrelationen die Quadratwurzeln der Fehler- und Konsequenzzahlen verwendet. Dadurch wird die rechte Seite der Verteilung gestaucht, wodurch auch mögliche Ausreißer, die das Korrelationsmaß erheblich beeinflussen können, etwas entschärft werden. Für jene Klassen, für die auch der Kolmogorov-Smirnov-Test auf Poissonverteilung signifikant ist (MAGNITUDE OF CONTROL, NOUN, SPACE MODELLING, STAGNATION, REPETITION und REGRESSION) wurde die Anwendbarkeit des Wurzelverfahrens nochmals geprüft. Dabei wurde entschieden, für STAGNATION, HELP, MAGNITUDE OF CONTROL und REPETITION die Quadratwurzelwerte zu verwenden, da die resultierende Variable einen höheren Signifikanzwert im K.-S.-Test auf Normalverteilung erreicht. Für den Nomenfehler war das Ergebnis umgedreht, weshalb hier die normalen Werte beibehalten wurden. Für REGRESSION und SPACE MODELLING war der Signifikanzwert für beide Varianten der Variablen gleich null, so dass eine Entscheidung auf anderen Kriterien basieren musste. Es wurde in beiden Fällen die Quadratwurzelvariante gewählt, da sie Ausreißer beinhalten. Bei PARTIAL PROGRESS schließlich wurde allein auf Grundlage des Histogramms entschieden, dass das Wurzelziehen die Verteilung nicht an die Gaussdistribution annähern würde.

Die Anforderungen an die Variablen für die Berechnung des Pearson-Koeffizienten, normalverteilt und mindestens intervallskaliert zu sein, kann nach diesem Ergebnis nicht als gegeben betrachtet werden. Dennoch wurde *r* berechnet, um von den oben genannten Vorteilen des Koeffizienten profitieren zu können. Es wurde jedoch zusätzlich Spearmans

ρ berechnet, um die Verlässlichkeit von *r* zu kontrollieren. Die in Kapitel 3.3.3 berichteten Korrelationen zeigen alle eine gemeinsame Tendenz für beide Koeffizienten[18].

3.3.1.2 Abhängige Fälle

Die obigen Erwägungen basieren auf der Auswertung aller Dialoge als separate Fälle. Dieses Vorgehen muss insofern kritisch beurteilt werden, als jeweils drei Dialoge von der selben Versuchsperson bewertet wurden, was bedeutet, dass diese Fälle nicht voneinander unabhängig sind und somit kein zufälliges Sample der Population darstellen. Die Unabhängigkeit der Fälle ist aber Voraussetzung für die Berechnung der Signifikanz des Ergebnisses der Korrelationsberechnung nach der üblichen Methode.

Will man dieses Problem umgehen, verbleiben zwei Möglichkeiten, den Datensatz zu analysieren. Entweder mittelt man die Variablen über alle Dialoge einer Versuchsperson, oder man analysiert die Daten in drei Gruppen, die jeweils nur einen Dialog jedes Probanden enthalten. Da die Verteilungen der Urteile eher der Gaussdistribution entsprachen, wenn der Datensatz nach den Metaphern geteilt wurde, erscheint dies auch für weitere Analysen vielversprechend. Die erste Möglichkeit birgt das Problem, dass über gute und schlechte Interaktionen bzw. Bewertungen einer Person gemittelt würde, wodurch interessante Informationen, nämlich über die Unterschiede in der Bewertung unterschiedlich gut verlaufener Dialoge, verloren gingen und sich möglicherweise sogar zu einem gewissen Grade gegenseitig aufheben würden. Die zweite Möglichkeit ist deshalb suboptimal, weil drei, unter Umständen relativ unterschiedliche Ergebnisse daraus resultieren, d. h., diese wären nur für einen Teil des Experiments gültig. Das schwerwiegendere Problem besteht aber darin, dass alle drei Ergebnisse aus einer sehr kleinen Anzahl von Fällen errechnet werden müssten. Es wurde also entschieden, vorerst das Problem der Abhängigkeit der Fälle in Kauf zu nehmen, da die anderen Varianten gewichtigere Kritikpunkte aufweisen.

[18] Diese Vorgehensweise findet sich auch bei Boland et al. 2004.

3.3.2 Faktorenanalyse der Benutzerurteile

Aus dem vorliegenden Datensatz mit 37 Benutzerurteilen aus den Interaktionsfragebögen und 19 Fehler- bzw. Konsequenzklassen ergeben sich 703 Paare von Merkmalen, deren Korrelation für die geplante Untersuchung interessant ist. Die seltenen Fehlerklassen sind dabei nicht mit eingerechnet. Da der Signifikanztest, der die Möglichkeit eines zufälligen Zustandekommens der Merkmalskonstellation misst, normalerweise mit $p<0,05$ (signifikant) bzw. $p<0,01$ (sehr signifikant) angesetzt wird, wären bei einer derart hohen Anzahl von Rechnungen dem Test zufolge bis zu 35 signifikante bzw. 7 sehr signifikante Korrelationen mit dem Zufall zu vereinen. Dazu muss gesagt werden, dass ohnehin der Signifikanztest durch die teilweise Abhängigkeit der Fälle in seiner Gültigkeit angegriffen wird. Da keine adäquate Regel für die Anpassung des Signifikanzniveaus gefunden wurde, wurde stattdessen die Zahl der sich ergebenden Korrelationen verringert, indem die Benutzerurteile zu Faktoren zusammengefasst wurden. Auch diese Arbeit wurde bereits im Rahmen des INSPIRE Projektes durchgeführt, so dass die Faktoren direkt aus Boland et al. (2004) übernommen werden konnten. Die Werte der einzelnen Faktoren wurden mit SPSS nach der in der Literatur angegebenen Methode neu berechnet, da sie in der Literatur nicht angeführt waren[19]. Details über das Verfahren sowie die Reliabilitätsstatistik sind in dem genannten Deliverable dokumentiert. Hier soll lediglich die Deutung der gefundenen Dimensionen wiedergegeben werden:

> *C1: The first dimension seems to be related to acceptability. Highest loadings are observed for the question whether the service would be used again, or whether a different interface would be preferred for carrying out the given tasks. System helpfulness, comfort and efficiency are related to this dimension, as well as being in control over the interaction. This dimension is the one with the closest relationship (highest loading) to user satisfaction (statement 6.5).*
>
> *C2: Dimension two describes the cognitive demand which is necessary to follow the dialogue, with high loadings on the required concentration and the stress (negative sign for relaxation) imposed on the user. Interestingly, there is hardly any relationship to the listening-effort, cf. dimension five.*
>
> *C3: This dimension is a task-efficiency-related component, with high loadings on task success, on the clarity of the provided information, and on the transparency of the system behaviour.*

[19] Die Ergebnisse sind exakt gleich, da auch die Berechnungen für das Projekt mit SPSS getätigt wurden.

C4: Dimension four is related to system errors, reflected by the frequency of system errors and the reliability of the system.

C5: Dimension five describes the ease of use, and has two high loadings: One on listening effort (negative sign), and one on the ease of learning.

C6: Dimension six describes the system's cooperativity, which is the only statement with a loading higher than 0.6 on this dimension.

C7: This dimension shows two high loadings which cannot be interpreted in combination: The naturalness of the system voice, and the symmetry of the dialogue. Both have a strong positive impact on this dimension.

C8: Dimension eight characterizes the speed of the interaction, with a moderate loading on the length of the dialogue as well.[20]

Die beschriebenen Faktoren unterscheiden sich deutlich von der Gruppierung der Fragen im Fragebogen, sind jedoch nicht untypisch für die Beurteilung von SDS. Dies wurde von Möller (2005) gezeigt, der die in verschiedenen Experimenten mit SDS gefundenen Dimensionen der Qualitätsurteile verglichen hat. Hier wurden insbesondere die Dimensionen ACCEPTABILITY, COMMUNICATION EFFICIENCY, COGNITIVE EFFORT und SMOOTHNESS als häufige und mit den INSPIRE Dimensionen relativ kongruente Faktoren identifiziert. Damit können die Faktoren als plausible Zielgrößen einer Qualitätsvorhersage betrachtet werden. Ferner hat eine systemübergreifende Bedeutung der Faktoren den Vorteil, dass anhand eines weiteren SDS, dessen Qualitätsbeurteilung ähnliche Dimensionen aufweist, untersucht werden könnte, inwieweit sich die Zusammenhänge zwischen den Qualitätsdimensionen und den Fehlerzahlen bei verschiedenen Systemen ähneln.

Für die Fehler ist eine Faktoranalyse nicht sinnvoll, da eine Dimension aus verschiedenen Phänotypen nicht gedeutet werden könnte. Gleich „aussehende" Fehler können sehr unterschiedliche Ursachen haben, die im komplexen Zusammenspiel der Benutzer- und Systemeigenschaften zu suchen sind. Korrelationen (als Grundlage der Faktorenanalyse) zwischen den Auftretenshäufigkeiten verschiedener Fehlerklassen können jedoch nur durch gemeinsame Ursachen und nicht durch ein gemeinsames Erscheinungsbild zustande kommen, da die Auftretenshäufigkeit offensichtlich nicht von letzterem abhängig ist.

[20] Möller (2005; Seiten nicht nummeriert)

Daher ließe eine Dimensionssuche auf Grundlage der phänotypischen Klassen, die ja nicht einmal einzeln genotypisch gedeutet werden können, kein sinnvolles Ergebnis erwarten.

Eine weitere Möglichkeit der Gruppierung, und damit der Reduzierung der Anzahl von Korrelationen, wäre die Zusammenfassung der Unterklassen zu den Hauptklassen, also CAPABILITY, VOCABULARY, STATE und MODELLING. Jedoch ist auch in diesem Fall zu erwarten, dass aufgrund der Heterogenität der Unterklassen untereinander die Korrelationseffekte verwischt würden. Deshalb wurden die Fehlerklassen nicht zusammengefasst und es wurden Korrelationen zwischen diesen und denn acht aus den Benutzerurteilen extrahierten Faktoren berechnet.

3.3.3 Korrelationen mit Fehlerzahlen

Für die 8 mal 19 gleich 152 interessanten Korrelationen wurden jeweils Spearmans ρ und Pearsons *r* berechnet. Um bei der Abschätzung der Signifikanz der gefundenen Zusammenhänge beiden Koeffizienten gerecht zu werden, wurde als Signifikanzkriterium $p<0,05$ für ρ wie für *r* angenommen. Der Signifikanztest wurde einseitig ausgeführt, da eine negative Korrelation zwischen den Fehlerzahlen und den Qualitätsurteilen (höherer Wert bedeutet positiveres Urteil) angenommen wurde, die Nullhypothese also lautete, dass kein negativer Zusammenhang existiert (sondern entweder ein positiver oder gar keiner). Diese Kriterien erfüllen 17 der 152 korrelierten Paare (11,2%).

Korrelationen INSPIRE (alle Metaphern)

Faktor 1 (Acceptability)	***r***	**ρ**
Unprogressive State (sqrt.)	-0,363799633	-0,294275348
Repetition (sqrt.)	-0,346895928	-0,218700373
Noun	-0,32996	-0,26118
Faktor 2 (Cognitive Demand)	***r***	**ρ**
No Input (sqrt.)	0,291129776	0,227336805
Unprogressive State (sqrt.)	-0,263776113	-0,257644127
Verb (sqrt.)	-0,217325071	-0,239232512

Faktor 5 (Ease of Use)	***r***	**ρ**
Reference (sqrt.)	0,27580883	0,272823405
Stagnation (sqrt.)	-0,34735542	-0,296924829
Repetition (sqrt.)	-0,300971795	-0,227663011
Help (sqrt.)	-0,26977	-0,2447
Faktor 6 (Cooperativity)	***r***	**ρ**
No Input (sqrt.)	0,304703382	0,340930153
Reference (sqrt.)	0,255639642	0,252920657
Noun	-0,22841	0,226718372
Faktor 7 (- unnamed -)	***r***	**ρ**
Regression (sqrt.)	0,314764552	0,226718372
Faktor 8 (Speed of the Interaction)	***r***	**ρ**
Attribute Type (sqrt.)	0,270215	0,314208
Reference (sqrt.)	-0,244644897	-0,301448748

Tabelle 3.1Signifikante Korrelationen für den gesamten Datensatz aus dem Free-Wizard-of-Oz-Experiment mit dem INSPIRE System.

Es sei hier noch kurz darauf hingewiesen, dass der hier beschriebene „Signifikanztest" streng genommen kein Signifikanztest im Sinne eines Tests auf die Irrtumswahrscheinlichkeit bzgl. der Annahme ist, der gefundene Zusammenhang sei zufällig. Die Bezeichnung eines Ergebnisses als „signifikant" bedeutet hier also nicht, dass das Ergebnis zuverlässig für die Gesamtpopulation angenommen werden kann. Die Ergebnisse beschreiben demnach nur den hier benutzten Datensatz selbst. Das Problem der „Generalisierbarkeit" wird auch im Zusammenhang mit den Regression Trees (Kapitel 5) vorhanden sein (siehe auch Compagnoni 2006). Bei der Auswertung der Koeffizientenwerte ist zu beachten, dass der Wert für ρ gegenüber *r* als der zuverlässigere zu betrachten ist, da die Voraussetzungen der Korrelationsanalyse nach Pearson nicht eingehalten wurden. Unter Berücksichtigung dieser Überlegungen soll im Folgenden eine Analyse der Ergebnisse versucht werden.

Interessant und kritisch zu bewerten ist die Tatsache, dass sechs der siebzehn gefundenen Werte positiv sind. Da bei der Übertragung der Urteile in die SPSS-Datei jeweils positive Antworten positive Werte zugeordnet bekamen, bedeutet eine positive Korrelation, dass in

diesen Fällen eine höhere Fehlerzahl mit einer besseren Beurteilung durch den Benutzer einherging. Es sollen zunächst Erklärungen für diese unerwarteten Ergebnisse versucht werden.

Einen relativ hohen (im Vergleich zu den anderen Betragswerten), positiven Wert weist die Korrelationen zwischen dem NO-INPUT-Fehler und dem COOPERATIVITY-Faktor auf (*r* = 0,304703 und ρ = 0,34093). Möglicherweise lässt sich dieses Ergebnis darauf zurückführen, dass das System bei Auftreten von NO INPUT normalerweise einen Hilfeprompt abspielt oder
im Falle des Start-GDN (Was kann ich noch für sie tun?), in dem ein Nutzer evtl. nicht weiß, wie er mit dem System sprechen soll, konkret nach dem Gerät fragt, das bedient werden soll.

Eine positive Korrelation besteht auch zwischen dem Auftreten des ATTRIBUTE-TYPE-Fehlers und dem Faktor SPEED OF THE INTERACTION (*r*= 0,270215; ρ= 0,314208). Der ATTRIBUTE TYPE ERROR trat häufig bei der Aufgabe auf, eine Nachrichtensendung zu wählen. Dabei kam es vor, dass der Sendungsname „heute“ als Tag „heute“ fehlinterpretiert wurde, was jedoch zufällig ein korrektes AVP war, das somit nicht mehr bestimmt werden musste. Bei der Nennung anderer Nachrichtensendungen, z. B. der Tagesschau, wurde diese teilweise auf eine ganze Reihe von AVPs gemappt, da der Name als Synonym für den Sendungstyp „Nachrichten“ verstanden wurde, was wiederum verschiedene andere AVPs wie z. B. Fernseher als Gerät impliziert. Dennoch muss hier gesagt werden, dass in der Regel auch ATTRIBUTE-TYPE-Fehler den Dialog unnötig verlängerten und häufig mit STAGNATION einhergingen.

Die positive Beziehung zwischen dem REFERENCE-Fehler und dem Faktor EASE OF USE (*r*= 0,275809; ρ= 0,272823) lässt sich möglicherweise damit erklären, dass die Benutzer, die diese Art von Fehler gemacht haben, weniger stark darauf geachtet haben, sich dem System als einem “dummen” Computer verständlich zu machen sondern auch linguistisch und semantisch komplexere Äußerungen ausprobierten. Schaut man sich den Faktor genau an, sieht man auch, dass eine der beiden Variablen, die ihn im wesentlichen konstituieren, nämlich „Ich musste mich konzentrieren, um das System akustisch zu verstehen“, negativ

lädt, so dass bei einer positiven Korrelation des Faktors zu dieser Variablen dennoch eine negative Beziehung besteht. Der REFERENCE-Fehler korreliert auch positiv mit dem COOPERATIVITY-Faktor (r= 0,25564; ρ= 0,252921), was die Annahme des eher zwanglosen Umgangs der Nutzer mit vielen REFERENCE-Fehlern insofern bestätigt, als die Einschätzung des Systems als kooperativ diesen zwanglosen Umgang motiviert haben könnte. Würde eine Versuchsperson das System andererseits als nicht kooperativ einschätzen, würde sie vermutlich versuchen, Fehler zu vermeiden, indem sie u. a. einfache Sätze bildet.

Erstaunlich ist, dass auch die Konsequenzklasse REGRESSION, die das Unheil quasi schon im Namen führt, mit einem Faktor positiv korreliert. Es handelt sich dabei um den unbenannten siebten Faktor, der die Urteile „Naturalness Of The System Voice“ und „Symmetry Of The Dialogue“ repräsentiert (r= 0,314765; ρ= 0,226718). Eine sinnvolle Verbindung dieser drei Merkmale lässt sich kaum finden, insbesondere die Natürlichkeit der Systemstimme sollte eigentlich mit keiner der Fehler- oder Konsequenzklassen korrelieren, da hier nicht die Interaktion bewertet wurde. Da auch der Faktor nicht erklärt werden konnte, liegt hier möglicherweise eine ungünstige Verteilung der Daten vor, wie z. B. ein Ausreißer, die die Korrelationswerte in die Höhe treibt.

Alle anderen, und damit die Mehrzahl der Korrelationen, die gefunden wurden, sind den Erwartungen entsprechend negativ. Wegen des mangelhaften Signifikanztests für die Werte sollen auch diese bzgl. ihrer Plausibilität reflektiert werden.

Der erste Faktor, ACCEPTABILITY, korreliert mit dem UNPROGRESSIVE-STATE-Fehler (r= -0,3638; ρ= -0,29428) und dem Nomenfehler (r= -0,32996; ρ= -0,26118), sowie mit der Konsequenz REPETITION (r= -0,3469; ρ= -0,2187). Da dieser Faktor vor allem generelle Statements zu dem System bzw. dessen Tauglichkeit für die Bedienung der Geräte enthält, wäre hier eigentlich eine Korrelation mit jedem der Fehler selbstredend erklärt. Alle drei Paare zeigen relativ hohe Werte für Pearsons r, die sich lediglich im Fall von REPETITION nicht in einem hohen ρ-Wert widerspiegeln.

Die Korrelation zwischen dem UNPROGRESSIVE-STATE-Fehler und dem Faktor COGNITIVE DEMAND (r= -0,26378; ρ= -0,25764) ist dagegen auffälliger, da die entsprechenden Urteile einen spezifischen Aspekt des Systems bzw. der Interaktion betreffen. Auch diese Korrelation erscheint jedoch sehr plausibel, da UNPROGRESSIVE-STATE-Fehler in der Regel dann auftauchten, wenn der Benutzer eher unsicher agierte, also z. B. im Zweifel schien, ob das System ihn korrekt verstanden habe. Die selbe Argumentation kann auch für die Korrelation des Faktors mit NO-INPUT-Fehlern (r= 0,29113; ρ= 0,227337) gelten. Beide Beziehungen sind im Vergleich zu den anderen gefundenen Korrelationen jedoch nur moderat bis schwach. Der COGNITIVE-DEMAND-Faktor korreliert außerdem noch mit der VERB-Fehlerklasse (r= -0,21733; ρ= -0,23923). Eine sinnvolle Begründung, warum der Faktor mit dem Verbfehler, aber z. B. nicht mit dem Nomenfehler einhergeht, kann vorerst nicht gegeben werden. Möglicherweise sind Synonyme für Verben schwerer zu finden als Synonyme für Nomen. Die Werte der Koeffizienten sind letztlich auch ziemlich niedrig, wobei auffällig ist, dass ρ betragsmäßig größer ist als r. Möglicherweise wird r hier dadurch gemindert, dass die Beziehung durch das Wurzelziehen bei der Vorbereitung der Daten nicht mehr linear ist.

Als nächstes sei die fünfte gefundene Dimension betrachtet. Der Faktor EASE OF USE korreliert relative deutlich mit der Konsequenz STAGNATION (r=-0,34736; ρ= -0,29692), sowie deren Spezialfällen REPETITION (r= -0,30097; ρ= -0,22766) und HELP (r= -0,26977; ρ= -0,2447). Hier ist überraschend, dass REPETITION und HELP annähernd in gleichem Maß mit der wahrgenommenen Einfachheit der Bedienung des Systems einhergehen, obwohl eigentlich durch die Hilfe-Prompts die Benutzung erleichtert werden sollte. Man kann andererseits aber auch argumentieren, dass dem Abspielen eines Hilfe-Prompts in der Regel eine problematische Situation vorangeht, in der entweder der Benutzer nach Hilfe fragt oder das System diese automatisch anbietet. Weiterhin wurde mehrmals ein Hilfeprompt durch einen Phrasenfehler ausgelöst, bei dem fälschlicherweise das „Help Request“-AVP extrahiert wurde, und einige Male wurde ein unpassender Hilfe-Prompt abgespielt, was offensichtlich eher Verwirrung stiftet als dass es die Bedienung erleichtert. Insgesamt kann man sagen, dass bei häufiger Stagnation des Dialogs ein wie immer geartetes negatives Urteil zu erwarten ist, jedoch wären Korrelationen mit anderen

Faktoren, z. B. SPEED OF THE INTERACTION, noch eher zu erwarten gewesen. Dennoch ist diese Korrelation die zweithöchste, die hier überhaupt gefunden wurde.

Der Faktor TASK EFFICIENCY zeigt eine schwache negative Beziehung zu NO-INPUT-Fehlern (r= -0,23871; ρ= -0,21698). Da NO-INPUT-Fehler meistens auf eine längere Stagnation des Dialogs folgten, ist dieser Zusammenhang durchaus plausibel. Der kausale Zusammenhang ist jedoch umgekehrt zu den erwarteten Zusammenhängen, d. h., der Benutzer verweigerte offensichtlich wegen der Einschätzung der Interaktion als wenig effizient für einen Moment die Kooperation mit dem System, indem er keine Eingabe machte.

Eine weitere Korrelation besteht zwischen der Dimension COOPERATIVITY und Nomenfehlern (r= -0,22841; ρ= -0,3157), wobei in diesem Fall ρ betragsmäßig deutlich höher ist als r. Dies ist sehr ungewöhnlich und deutet darauf hin, dass es sich um einen nichtlinearen Zusammenhang zwischen den Variablen handelt. Die Stärke des Zusammenhangs wird hier also mglw. durch die errechneten Werte unterschätzt. Diagramm 3.1 zeigt, dass bis zu drei Nomenfehlern die COOPERATIVITY-Werte fallen, dann jedoch wieder steigen, wobei der Anstieg nicht monoton ist. Dies könnte bedeuten, dass ab einer bestimmten Anzahl Fehler andere Einflüsse (z. B. eine einfache Behebung von Fehlern) bei der Beurteilung stärker ins Gewicht fallen. Bei Betrachtung nur der Fälle mit bis zu drei Fehlern kann man sagen, dass die Beziehung der Merkmale nicht linear ist, so dass mglw. dieser Teil der Daten für den höheren Wert für ρ verantwortlich ist.

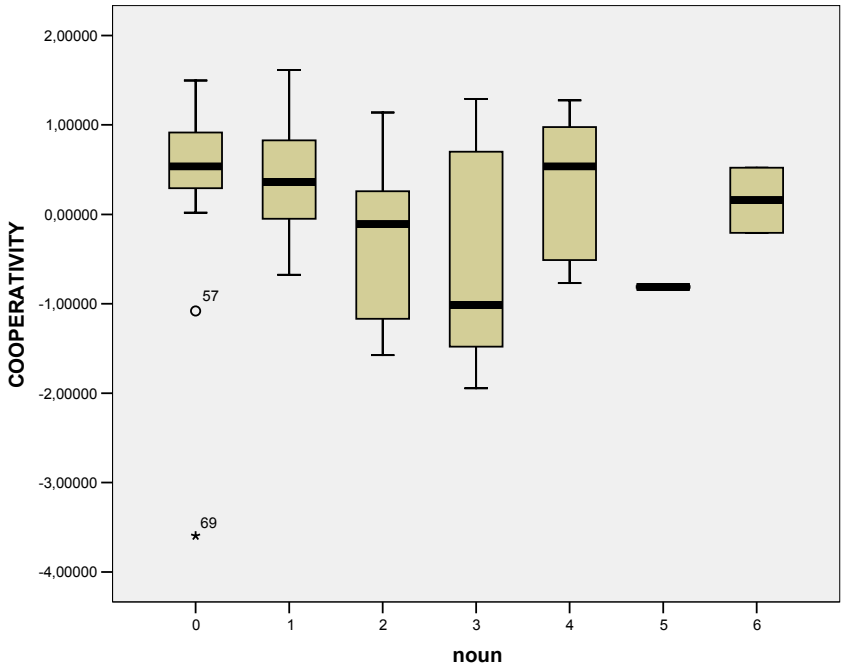


Diagramm 3.1Beziehung zwischen dem COOPERATIVITY-Faktor und Nomenfehlern. Bis drei Fehler fällt der Urteilsfaktor monoton, jedoch nicht linear. Bei mehr als drei Fehlern ist die Beziehung auch nicht mehr monoton.

Schließlich besteht eine letzte Korrelation zwischen SPEED OF THE INTERACTION und dem REFERENCE-Fehler (r= -0,24464; ρ= -0,30145). Prinzipiell ist der Zusammenhang gut nachvollziehbar, da Kommunikation durch das Referenzieren gemeinsamen Wissens deutlich vereinfacht wird. So müsste im Fall der von den Versuchspersonen durchgeführten Aufgaben zum Beispiel ein Film, der einmal ausgewählt wurde, nicht nochmals mittels des relativen langen Dialoges, der dazu notwendig ist, selektiert werden, wenn lediglich die betreffende Programmhandhabung nachträglich geändert werden soll. Das System kann in diesem Fall jedoch nicht mit der Geschwindigkeit des Benutzers bei der Kommunikation Schritt halten und wird deshalb von diesem als „langsam“ beurteilt.

3.3.4 Metaphernweise Analyse

Diese Ergebnisse sind aus mehreren Gründen nicht vollends befriedigend. Zum einen korreliert nur ein relativ geringer Anteil von 11% der Paarungen nach den gestellten Kriterien signifikant, obwohl ein relativ deutlicher Einfluss der Fehler auf die Beurteilung der Interaktionen angenommen werden kann. Zudem wird dieses Ergebnis durch eine relativ hohes signifikanzniveau von 5% relativiert. Weiterhin sind immerhin sechs der 17

errechneten Koeffizientenwerte positiv und müssen streng genommen von den signifikanten Ergebnissen ausgeschlossen werden, da auf einen negativen Zusammenhang spekulierend einseitig getestet wurde. Die Werte selbst bestätigen also die Nullhypothese, gegen die mit dem Signifikanztest getestet wurde. Weiterhin sind die Korrelationen relativ niedrig, obwohl Ausreißer bei einigen der Paare hohe Koeffizientenwerte begünstigen.

Jedoch lässt die Tatsache, dass fast alle gefundenen Zusammenhänge (inklusive der unerwarteten positiven Korrelationen) gut erklärt werden konnten, die Vermutung zu, dass durchaus Effekte vorhanden sind, die jedoch mglw. komplexere Analysemethoden erfordern. Da bei der Analyse der Daten im Rahmen des INSPIRE Projektes gezeigt wurde, dass sich die Verteilung der meisten Variablen der Gausskurve annähert, wenn die Daten nach der Metapher geteilt werden, wurde im Folgenden (trotz der weiter oben geäußerten Vorbehalte) der Versuch unternommen, mit dieser Methode auch stärkere und zuverlässigere Korrelationen zu finden.

Dafür wurde für jede Metapher zunächst eine Faktorenanalyse der Benutzerurteile gemacht. Zwar ist dieses Verfahren für die relativ geringe Anzahl an verbleibenden Fällen (n=24) kritisch zu beurteilen, jedoch wurde ein möglichst mit der vorherigen Analyse vergleichbares Ergebnis angestrebt. Um auch die Einzelergebnisse für die drei Metaphern vergleichen zu können, wurde ferner darauf geachtet, eine gute Übereinstimmung der Dimensionen zwischen den Metaphern zu erzielen. Letztendlich wurde je Metapher eine einheitliche Anzahl von vier Faktoren extrahiert, von denen die ersten beiden teilweise auf die selben Urteile laden. Folgende Tabelle zeigt die vier Faktoren mit den jeweiligen Einflussgrößen.

METAPHER 1	**METAPHER 2**	**METAPHER 3**
Erklärte Varianz: 64,5% Faktor 1	Erklärte Varianz: 65,6%	Erklärte Varianz: 62,3%
Pleasure and Worthwhileness of Interaction (25,6%) operation worthwhile (0,879) relaxation (0,861) interaction pleasant (0,794) dialogue short (0,774) use again (0,753) no concentration required (0,733) interaction fun (0,713) overall satisfied (0,691) dialogue symmetric (0,624) interface preferred (0,618) clear dialogue flow (0,606) dialogue smeeth (0,602)	**Acceptability (22,6%)** efficient handling (0,825) dialogue symmetric (0,789) system flexible (0,751) system helpful (0,732) operation comfortable (0,715) overall satisfied (0,673) operation worthwhile (0,658) use again (0,638) operation easy (0,630) dialogue short (0,609)	**Interface Qualification (17,8%)** interface preferred (0,858) efficient handling (0,828) use again (0,826) overall satisfied (0,717) operation worthwhile (0,711) operation comfortable (0,650)
Faktor 2		
Communication with the System (19,6%) perceived understanding (0,918) overall quality (0,807) operation easy (0,782) information complete (0,666) task success (0,664) system realiable (0,637) error recovery (0,605) system helpful (0,601)	**Communication Success (16,7%)** task success (0,882) overall quality (0,873) perceived understanding (0,812)	**System Behaviour (16,2%)** system flexible (0,810) operation easy (0,791) system realiable (0,790) interaction fun (0,743) interaction pleasant (0,626) system cooperative (0,604)
Faktor 3		

System Transparency (10,9%) dialogue fast (0,798) reaction as expected (0,707) system transparent (-0,619) information clear (0,611) no listening effort (-0,603)	**Cooperativity (13,4%)** system cooperative (0,791) reaction as human (0,733) no listening effort (-0,636) reaction as expected (0,625)	**System Transparency (15,7%)** syst. expectation known (0,782) information clear (0,672) clear dialogue flow (0,672) error recovery (0,657) no concentration required (0,621)
Faktor 4		
Cooperativity (8,3%) reaction as human (0,859) system cooperative (0,771)	**User Relaxation (12,9%)** no concentration required (0,819) relaxation (0,797) system friendly (0,670) few errors (0,611)	**Efficiency of Communication (12,7%)** task success (0,709) system fast (0,655) reaction as human (0,643) reaction as expected (0,634) dialogue short (0,603)

Tabelle 3.2 Faktoren für die drei Systemmetaphern des INSPIRE-Experiments

Für jede Metapher wurden anschließend die Verteilungen der Fehlerzahlen unter Zuhilfenahme des Kolmogorov-Smirnov-Tests mit den Verteilungen der radizierten Fehlerzahlen verglichen. Jeweils jene Variante, welche den höheren exakten Signifikanzwert aufwies, wurde für die Korrelationsberechnung gewählt. Bei gleichen Signifikanzwerten wurde zu Gunsten der Ausgangsvariable entschieden. Auch die Faktoren wurden auf ihre Verteilungseigenschaften getestet, wobei für einige Nicht-Normalverteilung diagnostiziert wurde. Hier konnte jedoch keine geeignete Transformation der Daten gefunden werden, um die Verteilungen anzupassen. Daher sollte die Nichteinhaltung der Voraussetzungen für die Berechnung von *r* bei der Auswertung wieder im Hinterkopf behalten werden.

Als Signifikanztest wurde auch hier eine Irrtumswahrscheinlichkeit unter 5% sowohl für den Pearson- als auch den Spearman-Koeffizient angenommen (einseitiger Test). Die

somit selektierten Korrelationen sind deutlich höher als beim kompletten Datensatz ($0.4 < |r| < 0.7$), was sich auch auf die geringere Anzahl an Fällen zurückführen lässt, wie die Formel für *r* zeigt.

$$r = \frac{\sum_i (M_i - \overline{M})(P_i - \overline{P})}{\sqrt{\sum (M_i - \overline{M})^2}\sqrt{\sum (P_i - \overline{P})^2}}$$

M_i und P_i sind hier die Werte der korrelierten Variablen für den Fall i. Während im Zähler die Summanden positiv und negativ sein können, werden im Nenner nur positive Werte summiert, so dass dieser schneller wächst als der Zähler. Je mehr Summanden (bzw. Fälle) es also gibt, desto kleiner wird der Wert *r*. Die Anzahl der Fälle wird jedoch auch beim Signifikanztest berücksichtigt, so dass unrealistisch positiven Deutungen der Zahlen vorgebeugt wird. Im Folgenden sollen die Ergebnisse für die einzelnen Systemmetaphern im Detail geschildert werden.

3.3.4.1 Metapher 1

Die Datenvorbereitung für die Analyse der Fälle mit Metapher 1 beinhaltete das Wurzelziehen der Werte für die Konsequenzen STAGNATION, REPETITION, HELP und REGRESSION. Von den sich ergebenden Korrelationen wurden zehn nach der beschriebenen Methode als signifikant eingestuft, was bei insgesamt 72 Paaren 13,9% entspricht, also einem größeren Anteil als bei der Analyse aller Fälle zusammen. Alle signifikanten Korrelationen entfallen auf die ersten beiden Faktoren PLEASURE AND WORTHWHILENESS OF INTERACTION und COMMUNICATION WITH THE SYSTEM.

Korrelationen INSPIRE Metapher 1		
Faktor 1 (Pleasure)	***R***	**ρ**
No Input	0,604646	0,525341
Unprogressive State	-0,60636	-0,39624
Time Modeling	0,462264	0,400694
Faktor 2 (Communication)	***R***	**ρ**

Magnitude of Control	-0,59129	-0,51083
Noun	-0,60512	-0,57005
Space Modelling	-0,44123	-0,4879
Reference	0,477702	0,493091
Stagnation (sqrt.)	-0,57424	-0,53345
Repetition (sqrt.)	-0,61457	-0,55185
Help (sqrt.)	-0,5134	-0,48421

Tabelle 3.3 Signifikante Korrelationen für Metapher 1 des Free-Wizard-of-Oz-Experiments mit dem INSPIRE-System.

Der erste Faktor, PLEASURE, korreliert mit NO-INPUT- und TIME-MODELLING-Fehlern positiv (r= 0,604646; ρ= 0,525341; bzw. r= 0,462264; ρ= 0,400694). Für dieses Ergebnis kann ad hoc keine Erklärung gefunden werden, da NO-INPUT-Fehler eigentlich meistens mit eher stressigen Situationen während der Interaktionen einhergehen und TIME-MODELLING-Fehler sich in diesem Experiment im Wesentlichen darauf beschränkten, einen Zeitpunkt mit „jetzt“ zu benennen, was kein Grund sein kann, den Dialog als angenehm zu empfinden.

Eine negative Korrelation hat dieser Faktor mit dem UNPROGRESSIVE-STATE-Fehler (r= -0,60636; ρ= -0,39624), wobei der relativ große Unterschied zwischen den beiden Koeffizienten verdächtig ist. Tatsächlich macht das Streudiagramm der Beziehung deutlich, dass es einen deutlichen Ausreißer in der unteren rechten Ecke gibt, auf den der Spearmansche Koeffizient nicht so empfindlich reagiert wie der Pearsonsche.

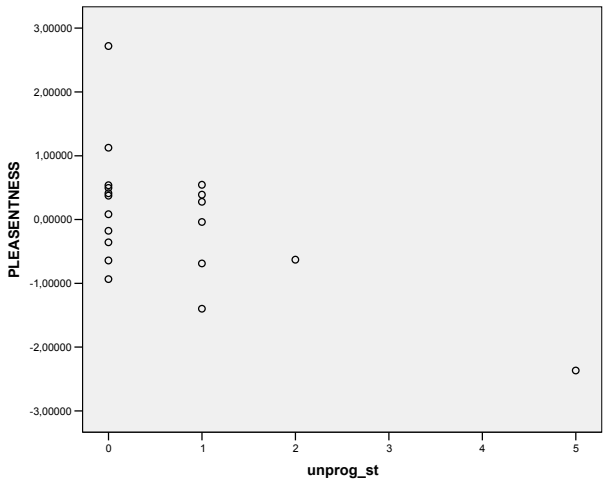


Diagramm 3.2 Streudiagramm der Beziehung zwischen UNPROGRESSIVE-STATE-Fehler und dem Faktor PLEASURE (Pleasantness) mit Ausreißer in der unteren rechten Ecke.

Der zweite Faktor, COMMUNICATION WITH THE SYSTEM, korreliert positiv mit dem REFERENCE-Fehler (r=0,477702; ρ=0,493091), was prinzipiell recht plausibel ist, da REFERENCE-Fehler, wie bereits in Kapitel 3.3.3 angedeutet wurde, auf einen eher ungezwungenen, jedoch aus Sicht des Systems relativ komplexen Sprechstil schließen lassen. Möglicherweise wurde mit der positiven Beurteilung der Kommunikation die Fähigkeit des Systems, einige andere komplexe Äußerungen korrekt zu verarbeiten oder die mit diesem Fehler behafteten Äußerungen zumindest teilweise zu verstehen, gewürdigt.

Neben dieser positiven gibt es eine Reihe von negativen Korrelationen. Sehr plausibel erscheint die Korrelation des Faktors mit der Konsequenz STAGNATION (r=-0,57424; ρ=-0,53345) und deren Sonderfällen REPETITION (r=-0,61457; ρ=-0,55185) und HELP (r=-0,5134; ρ=-0,48421), da diese Konsequenzen normalerweise dann auftreten, wenn der Benutzer überhaupt nicht verstanden wurde. Ein weiteres kommunikationsbezogenes Manko von Stagnations-Situationen ist, dass der Benutzer kein Feedback, auch nicht implizit, bekommt, warum das System nicht in den nächsten Zustand übergeht.

Auch der Nomenfehler, der üblicherweise in relativ einfachen Satzkonstruktionen auftrat, kann bzgl. seiner Korrelation mit dem COMMUNICATION-Faktor (r=-0,60512; ρ=-0,57005) gut gedeutet werden, indem die Argumentation für den REFERENCE-Fehler umgekehrt wird. Dass SPACE-MODELLING-Fehler, die wie REFERENCE-Fehler auf komplexeren Äußerungen beruhen, weniger stark korrelieren (r=-0,44123; ρ=-0,4879) passt gut in dieses Bild. Die entsprechenden Äußerungen sind meistens relativ lang, und dass das System in der Lage ist, Beziehungen zwischen den Objekten im Raum herzustellen, wie z. B., dass die Stehlampen neben dem Sofa stehen, mag auch eine laienhafte Testperson beeindruckt haben. Insgesamt sollte jedoch bei dieser Argumentation bedacht werden, dass nicht ohne weiteres davon ausgegangen werden kann, dass der Benutzer (und Beurteiler) sich über die Schwierigkeiten des Systems mit bestimmten Kategorien von Äußerungen im Klaren ist.

Schließlich besteht noch eine hohe Korrelation zwischen dem COMMUNICATION-Faktor und INADEQUATE-MAGNITUDE-OF-CONTROL-Fehlern (r=-0,59129; ρ=-0,51083). Dies

macht durchaus Sinn, da diese Fehler in der Praxis häufig nicht darauf beruhen, dass der Benutzer ein falsches Ziel hat, sondern darauf, dass das System das Ziel des Benutzers missverstanden hat.

3.3.4.2 Metapher 2

Korrelationen INSPIRE Metapher 2		
Faktor 2 (Communication Success)	***r***	**ρ**
Unprogressive State	-0,48331	-0,44781
Attribute Type	0,412894	0,423231
Repetition	-0,52223	-0,4389
Faktor 3 (Cooperativity)	***r***	**ρ**
Reference	0,450468	0,470589

Tabelle 3.4 Signifikante Korrelationen für Metapher 2 des Free-Wizard-of-Oz-Experiments mit dem INSPIRE-System.

Vor den zu Metapher 2 durchgeführeten Berechnungen wurden von INADEQUATE-MAGNITUDE-OF-CONTROL- und STATE-Fehlerzahlen, sowie von den Konsequenzen HELP und REGRESSION die Quadratwurzeln gezogen. Die Korrelationsanalyse fällt hier wesentlich magerer aus als bei Metapher 1: es wurden nur vier für beide Koeffizienten signifikante Korrelationen gefunden (5,5% von 72 Paaren), was bei einer Irrtumswahrscheinlichkeit von 5% auch auf den Zufall zurückgeführt werden könnte. Dennoch können erklärbare Korrelationen auch von Bedeutung sein, weshalb ein Deutungsversuch unternommen werden soll.

Wieder ergeben sich hier positive und negative Korrelationen. Positiv korrelieren Attributfehler mit dem Faktor COMMUNICATION SUCCESS (r= 0,412894; ρ= 0,423231) sowie REFERENCE-Fehler mit dem Faktor COOPERATIVITY (r= 0,450468; ρ= 0,470589). Die Gründe für einen positiven Wert für die REFERENCE-Fehlerklasse wurden bereits in den beiden vorangehenden Analysen erläutert. Eine positive Beziehung zwischen ATTRIBUTE-TYPE-Fehlern und der COMMUNICATION SUCCESS benannten Dimension kann dagegen nicht so einfach begründet werden. Ein Blick auf die Streudiagramme beider

Beziehungen zeigt jedoch, dass beide Korrelationen realistisch sind, d. h., nicht durch einzelne, extreme Sonderfälle verfälscht werden. Die errechneten Werte gehören unter den metapherweise berechneten zu den kleinsten.

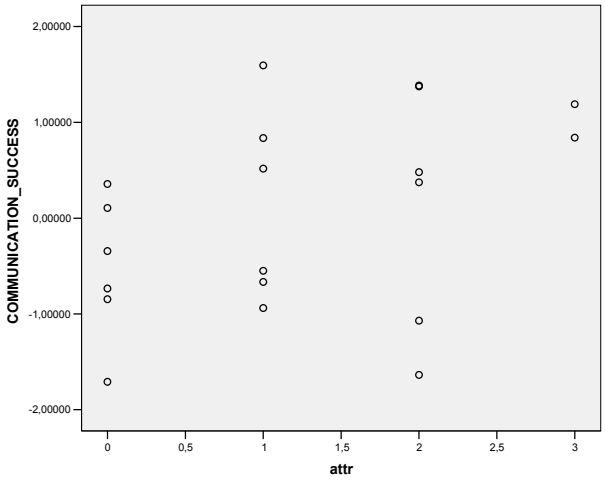


Diagramm 3.3 Streudiagramm der Beziehung zwischen ATTRIBUTE-TYPE-Fehler und dem Faktor COMMUNICATION SUCCESS. Tendentiell tauchen für höhere Werte der einen Variablen auch höhere Werte der anderen auf.

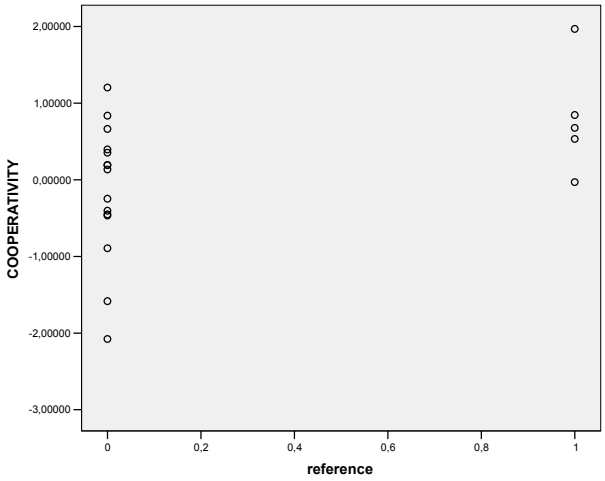


Diagramm 3.4 Streudiagramm der Beziehung zwischen REFERENCE-Fehler und dem Faktor COOPERATIVITY. Tendentiell tauchen für höhere Werte der einen Variablen auch höhere Werte der anderen auf.

Die beiden anderen Korrelationen haben ein negatives Vorzeichen und treten zwischen COMMUNICATION SUCCESS und dem UNPROGRESSIVE-STATE-Fehler (r= -0,48331; ρ= -0,44781) sowie der Konsequenz REPETITION (r= -0,52223; ρ= -0,4389) auf. Bei letzterer ist der Zusammenhang eindeutig: Wurde der vorherige Prompt wiederholt, hatte das

System die Äußerung nicht verstanden, auch nicht teilweise. Je häufiger der Dialog also stagnierte, desto weniger erfolgreich verlief die Kommunikation. Der Zusammenhang mit UNPROGRESSIVE-STATE-Fehlern lässt sich mglw. damit erklären, dass der Benutzer in einigen Fällen *gerade deshalb den Fehler begeht*, weil er *denkt*, das System habe ihn nicht verstanden. Es handelt sich bei dieser Erklärung jedoch um Spekulation, da lediglich aus den transkribierten Dialogen auf das Wissen des Nutzers geschlossen werden kann.

3.3.4.3 Metapher 3

Auch für Metapher 3 ergeben sich nur wenige Korrelationen, die nach den beschriebenen Kriterien signifikant sind (für SPACE MODELLING, STAGNATION und HELP wurden die Quadratwurzeln der Werte verwendet). Wie bei Metapher 1 beschränken sich diese auf die ersten beiden Faktoren, INTERFACE QUALIFICATION und SYSTEM BEHAVIOUR.

Korrelationen INSPRE Metapher 3		
Faktor 1 (Interface)	***r***	**ρ**
Regression	0,456075	0,399405
Partial Progress	0,442475	0,41868
Faktor 2 (Behaviour)	***r***	**ρ**
State	-0,65586	-0,60022
Repetition	-0,48799	-0,4987
Stagnation (sqrt.)	-0,5103	-0,48326

Tabelle 3.5 Signifikante Korrelationen für Metapher 3 des Free-Wizard-of-Oz-Experiments mit dem INSPIRE-System.

Von den fünf signifikanten Beziehungen sind wiederum zwei positiv, wobei diesmal die positiven Korrelationen auf den ersten Faktor, der die Qualifikation des Systems als Interface zur Steuerung der genutzten Geräte beschreibt, fallen. Die erste der beiden positiven Korrelation tritt mit der Konsequenz REGRESSION auf (r= 0,399405; ρ= 0,456075), was eigentlich absurd klingt, wenn man nur die Namen der Variablen betrachtet. De facto muss man aber davon ausgehen, dass der Benutzer den Rückschritt teilweise nicht wahrnimmt. So kann sich dieser unter anderem darin äußern, dass ein GDN eingeleitet wird, in dem zwischen zwei konkurrierenden Werten für das selbe Attribut

entschieden werden muss. Diese Beobachtung bestätigt sich in der Tatsache, dass der Sonderfall RESTART, bei dem die Regression deutlich zutage tritt, nicht im selben Maße korreliert. Das Streudiagramm zeigt, dass die Koeffizientenwerte nicht durch ungünstige Verteilungseigenschaften der Merkmale verfälscht werden.

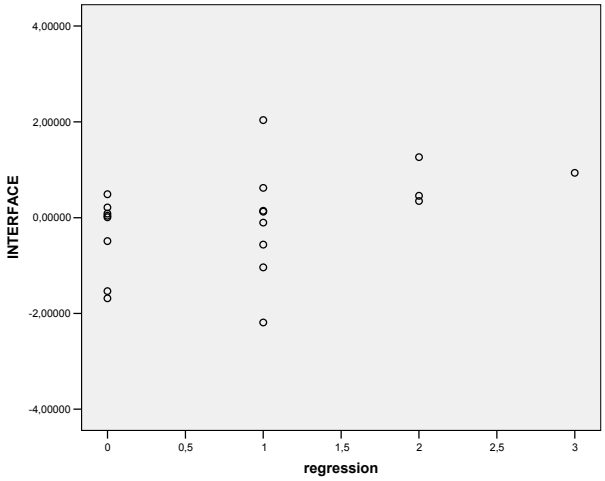


Diagramm 3.5 Streudiagramm der Beziehung zwischen der Konsequenz REGRESSION und dem Faktor INTERFACE QUALIFICATION. Eine leichte Tendenz von links unten nach rechts oben ist erkennbar.

Die zweite positive Korrelation entfällt auf die Konsequenz PARTIAL PROGRESS (r= 0,41868; ρ= 0,442475). Obwohl für diese Konsequenz nicht unbedingt ein explizit positiver Zusammenhang erwartet wurde, ist dieser prinzipiell sinnvoll und deutet darauf hin, dass ein (lediglich) partielles Voranschreiten des Dialogs nicht als Beeinträchtigung wahrgenommen wird.

Die drei negativen Korrelationen fallen auf den zweiten Faktor, dessen Einflussgrößen das Verhalten des Systems beschreiben. Eine sehr hohe Korrelation, sowohl für r als auch für ρ, ergibt sich mit dem STATE-Fehler (r= -0,60022; ρ= -0,65586). Das Auftreten eines solchen Fehlers, also Äußerung eines Kommandos, das gültig und progressiv wäre, wenn das System in einem anderen Zustand (GDN) wäre, bedeutet häufig partielles Voranschreiten des Dialogs (z. B. „Sollen die Nachrichten wirklich gelöscht werden?“ – „Ja, und schalte den Fernseher ein“; hier wird das „Ja“ verstanden). Im Sinne des vorherigen Absatzes spräche dies also gegen eine negative Korrelation.

Jedoch wurde auch ein STATE-Fehler annotiert, wenn der Benutzer die Hilfe anforderte, der passende Prompt aber zu einem anderen als dem aktuellen GDN gehörte. In diesen Fällen wurde ein inadäquater Hilfe-Prompt abgespielt, was ein relativ schwerer Fehler ist, wie schon weiter oben bemerkt wurde. Möglicherweise sind diese Situationen verantwortlich für die betragsmäßig hohe, negative Korrelation, zumal der Faktor BEHAVIOUR Urteile wie „das System ist zuverlässig“ oder „die Benutzung des Systems ist einfach“ repräsentiert.

Die anderen beiden signifikanten Korrelationen betreffen die Konsequenz STAGNATION (r= -0,48326; ρ= -0,5103) und deren Spezialfall REPETITION (r= -0,4987; ρ= -0,48799). Hier wird der Zusammenhang wohl unmittelbar deutlich: wenn das System häufig nicht auf die Benutzeräußerung reagiert, wird sein Verhalten (Kooperativität, Zuverlässigkeit etc.) schlechter eingestuft.

3.3.5 Analyse der seltenen Fehler

In der bisherigen Diskussion der Zusammenhänge zwischen Fehler- bzw. Konsequenzzahlen und Benutzerurteilen wurden nur jene Fehler berücksichtigt, für die sich rechnerisch ein Korrelationskoeffizient ermitteln lässt. Die Fehlerklassen, die zu selten auftraten, um sinnvoll durch eine derartige Gleichung beschrieben werden zu können, sollen im Folgenden wie angekündigt anhand der Streudiagramme auf entsprechende Beziehungen untersucht werden.

Es kann bereits im Vorhinein vermutet werden, dass entsprechende Zusammenhänge rar sind und relativ schwach, da diese Fehler aufgrund ihres seltenen Auftretens bei der Beurteilung relativ wenig ins Gewicht fallen. Es ist jedoch denkbar, dass ein seltener Fehler besonders schwerwiegende Konsequenzen hatte und deshalb trotz seiner Seltenheit urteilsprägend wirkte. Deshalb sollen diese Klassen nicht einfach übergangen werden. Für die Analyse sollen die dialogweisen Fehlerzahlen, farblich nach Metaphern unterschieden, gegen die acht Faktoren von Möller (2005) geplottet werden, sofern die Fehler auf unterschiedliche Metaphern verteilt sind.

Für den INEXISTENT-OBJECT-OF-CONTROL-Fehler trifft dies nicht zu. Er kommt nur einmal vor, und zwar in Metapher drei. Diagramm 5.5 zeigt die Beziehungen zu den acht Faktoren. Lediglich für den nicht erklärten Faktor (7) ist der dem Fehlerhaften Dialog zugeordnete Wert nicht durchschnittlich, es besteht jedoch eine positive Beziehung. Aus diesem einzelnen Wert auf einen kausalen Zusammenhang mit den Urteilen über Dialogsymmetrie und Natürlichkeit der Systemstimme zu schließen, ist wohl etwas zu gewagt, deshalb soll der INEXISTENT-OBJECT-OF-CONTROL-Fehler als bedeutungslos für die Beurteilung der Dialoge eingestuft werden.

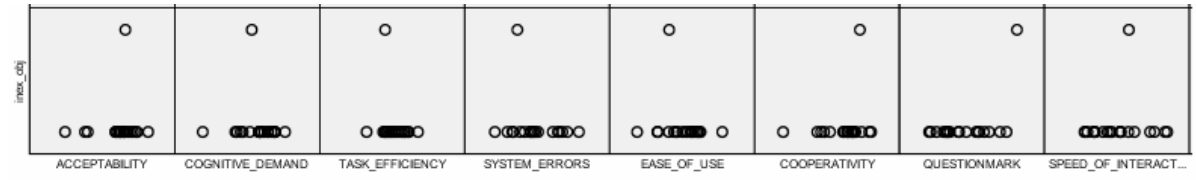


Diagramm 3.6 Streudiagramme von INEXISTENT OBJECT OF CONTROL Fehler über den acht Faktoren aus Möller (2005)

Der CONTROL-CONFLICT-(BECAUSE OF EXTERNAL RESTRICTIONS)-Fehler wurde drei mal annotiert, jedoch fehlen für eine der betroffenen Versuchspersonen die Urteile über die Gesamtqualität und damit auch die Faktorwerte. Deshalb sind in Diagramm 3.7 nur zwei Instanzen des Fehlers dargestellt. Auch hier lässt sich kein deutlicher Zusammenhang finden, lediglich bei EASE OF USE im Fall von Metapher 2 (grün) ist der zum fehlerhaften Dialog korrespondierende Faktorwert der niedrigste, so dass eine negative Beziehung angedeutet wird. Alle anderen Fehlervorkommen schlagen sich in durchschnittlichen Faktorwerten nieder.

Diagramm 3.7 Streudiagramme von CONTROL CONFLICT (BECAUSE OF EXTERNAL RESTRICTIONS) Fehler über den acht Faktoren aus Möller (2005); Metapher 1 ist blau, M. 2 grün und M. 3 gelb dargestellt.

Die Fehlerklasse ADJECTIVE/ADVERBIALE BESTIMMUNG wurde fünf mal annotiert, wobei in einem Dialog der Fehler gleich zwei mal gemacht wurde. Eine Instanz des Fehlers ist in

Diagramm 3.8 nicht dargestellt, da die Faktorwerte für diesen Dialog aufgrund einer fehlenden Fragebogenseite nicht berechnet werden konnten.

Für Metapher 1, blau dargestellt, kann in der Gegenüberstellung mit ACCEPTABILITY und COOPERATIVITY, sowie in gewissem Maße mit COGNITIVE DEMAND (nur eine Versuchsperson ohne Adjektivfehler bewertete in dieser Dimension positiver als der Teilnehmer mit Adjektivfehler), eine schwache positive Korrelation vermutet werden. Eine positive Korrelation ist jedoch sehr unwahrscheinlich, die Evidenz dafür hier hingegen nur sehr schwach.

Für Metapher 3 läßt sich bei der Beziehung zu ACCEPTABILITY eine relativ deutliche, negative Tendenz feststellen, da die Faktorwerte beider fehlerbehafteten Dialoge sehr niedrig sind, während die der fehlerfreien fast alle höher sind. Für TASK EFFICIENCY und den unbenannten siebten Faktor (im Diagramm: Questionmark) ist der Dialog mit zwei Fehlern vergleichsweise niedrigwertig vertreten, der Dialog mit einem Fehler teilt diese Tendenz jedoch nicht. Für COOPERATIVITY liegt die selbe Beziehung mit umgedrehten Vorzeichen vor.

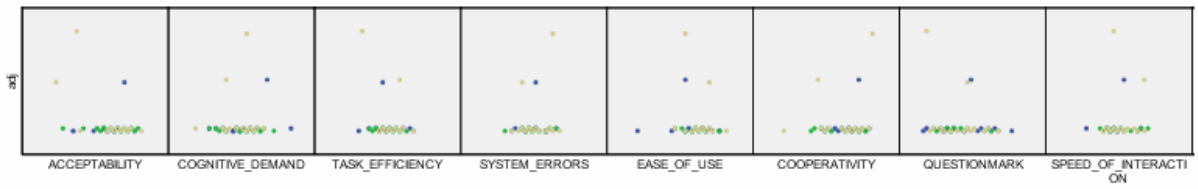


Diagramm 3.8 Streudiagramme des Fehlers ADJECTIVE/ADVERBIALE BESTIMMUNG über den acht Faktoren aus Möller (2005); Metapher 1 ist blau, M. 2 grün und M. 3 gelb dargestellt.

Sonstige Vokabelfehler (VOCABULARY (OTHER)) wurden zwei mal annotiert. Extreme Faktorenwerte für die entsprechenden Dialoge gibt es beim unbenannten Faktor und COOPERATIVITY, sowie für Metapher 1 bei TASK EFFICIENCY. Da von dem Klassennamen nicht auf die Art des Fehlers geschlossen werden kann, macht ein Blick in die annotierte Datei Sinn. Dabei kann einer der Fälle als Annotationsfehler identifiziert werden (es handelt sich eigentlich um einen Phrasenfehler). Die verbleibende Äußerung lautet „Ventilator ein ausschalten“, was vermutlich auch von dem betreffenden Benutzer als

unverständliche Äußerung erkannt wurde, sich also eher nicht auf sein Qualitätsurteil auswirkte. Demnach kann die Korrelationsanalyse für diese Klasse übergangen werden.

SYSTEM/WIZARD-OF-OZ-Fehler tauchen bei Metapher 2 und 3 zusammen sechs mal auf. Ein Fall wird hier nicht mitanalysiert, weil, wiederum wegen einer fehlenden Fragebogenseite, keine Faktorwerte errechnet werden konnten. Die Fehler dieser Klasse bestehen allesamt darin, dass das System eine korrekte Äußerung nicht versteht, z. B. aufgrund eines Tippfehlers des Wizards, und haben STAGNATION als Konsequenz. Möglicherweise wurden die betreffenden Versuchspersonen in einigen Fällen auf das technische Problem hingewiesen, da sie die jeweiligen Äußerung teilweise exakt wiederholten.

Extreme Faktorwerte weisen TASK EFFICIENCY (positive Beziehung) und der unbenannte Faktor (negative Beziehung) auf, und zwar für beide Metaphern. Obwohl auch fehlerfreie Dialoge ähnlich niedrige Faktorwerte zeigen, sollte hier eine kausale Beziehung nicht ausgeschlossen werden.

Diagramm 3.9 Streudiagramme des SYSTEM/WIZARD-OF-OZ-Fehler über den acht Faktoren aus Möller (2005); Metapher 1 ist blau, M. 2 grün und M. 3 gelb dargestellt.

Auch die nicht klassifizierten Fehler (OTHER-Klasse) sollen noch auf Korrelation mit den Faktoren untersucht werden. Von den sechs Fällen dieser Kategorie fällt einer wegen des fehlenden Gesamturteils weg. Von den verbleibenden Fehlern entfallen zwei auf Metapher 1, wobei diese beiden im Vergleich mit den fehlerfreien Dialogen für den Faktor COOPERATIVITY relativ hohe Werte aufweisen.

Einer dieser Fälle bestand in der Äußerung „die Lampe heller oder dunkler stellen“, worauf das System antwortete „Die linke Lampe ist bereits maximal hell. Was kann ich für Sie tun?“ (der Standort „links“ war bereits in einem früheren Systemzustand erfasst

worden). Für eine positive Beurteilung der Kooperativität spricht in diesem Fall, dass INSPIRE immerhin eine sinnvolle Bedeutung aus der offensichtlich unsinnigen Benutzeräußerung extrahiert. Dennoch hätte es idealerweise erkennen können, dass der Benutzer die Lampe herunterdimmen will, da auch dieses Konzept in der Äußerung auftauchte und pragmatisch gesehen plausibler gewesen wäre. Ein positiver Zusammenhang mit der Kooperativität des Systems kann also nur schwach begründet werden.

Der zweite „sonstige“ Fehler hätte eigentlich kategorisiert werden können, zählt also in Wirklichkeit gar nicht zu dieser Klasse und soll deshalb nicht weiter beachtet werden.

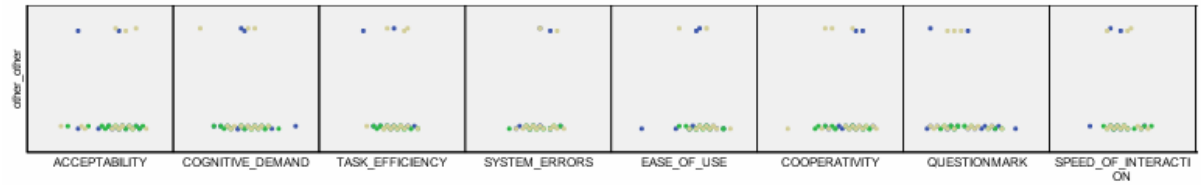


Diagramm 3.10 Streudiagramme der nicht klassifizierten Fehler über den acht Faktoren aus Möller (2005); Metapher 1 ist blau, M. 2 grün und M. 3 gelb dargestellt.

3.3.6 Zusammenfassung

Wegen der sehr hohen Anzahl von Korrelationen, ungünstiger Verteilungseigenschaften der Fehlerzahlen und Urteilswerte, sowie des Problems der Abhängigkeit jeweils dreier Fälle voneinander, wurden für die Bestimmung von Korrelationen und deren Signifikanz verschiedene, auf Ausgleich zielende Veränderungen des Datensatzes vorgenommen. So wurde die Anzahl der Korrelationen durch Reduktion der Urteile auf wesentliche Dimensionen (Faktoren) verringert. Durch die Verwendung von Quadratwurzelwerten wurde die Verteilung einiger Fehlerklassen der Normalverteilung angenähert. Zudem wurde als Signifikanztest ein signifikantes Ergebnis für den parametrischen Pearson-Koeffizienten sowie den Spearman-Koeffizienten, der für Ordinalskalen und nicht normal verteilte Daten angewendet werden kann, angenommen. Auf diese Art wurden relativ wenige signifikante Korrelationen mit niedrige Beträgen gefunden ($0{,}21 < |r| < 0{,}36$ bzw. $0{,}21 < |\rho| < 0{,}31$), die jedoch gut begründet werden konnten.

Um deutlichere Ergebnisse zu erhalten, wurde der Datensatz nach den Metaphern (Systemversionen) aus dem Free-Wizard-of-Oz-Experiment geteilt und getrennt analysiert. Dieser Ansatz war viel versprechend, da bekannt war, dass die Verteilungen der Urteile bei dieser Aufteilung sich der Gaussverteilung annähern (Boland et al. 2004) und so gleichzeitig das Problem der Abhängigkeit einiger Fälle umgangen wurde. Für jede Metapher wurden vier Faktoren der Urteile bestimmt, von denen die ersten beiden für alle drei Metaphern ähnliche Einflussgrößen aufwiesen. Die Korrelationen mit den Fehlerzahlen fielen deutlich höher aus als bei Betrachtung des kompletten Datensatzes ($0{,}39 < |r| < 0{,}62$ bzw. $0{,}31 < |\rho| < 0{,}66$), jedoch waren wiederum relativ wenige der Korrelationen für r und ρ signifikant. Diese konnten jedoch plausibel gemacht werden. Dabei zeigen die Ergebnisse der verschiedenen Analysemethoden eine gewisse Konsistenz. So korrelierten z. B. REFERENCE-Fehler häufig positiv mit der Dimension, die die Kommunikation mit dem System erfasst. Weiterhin wiesen die Konsequenz STAGNATION und deren Sonderfälle REPETITION und HELP mehrfach hohe, negative Korrelationen mit Faktoren auf.

3.4 Anpassung des Klassifikationsschemas

3.4.1 Kritik des bisherigen Fehlerklassifikationsschemas

Die Analyse der INSPIRE-Dialoge bestätigt also die Vermutung, dass es Zusammenhänge zwischen Fehlern und Beurteilungen einer Interaktion mit dem System gibt. Jedoch wären noch deutlichere Ergebnisse zu erwarten gewesen, angesichts dessen, dass ein fehlerfreier Dialog leicht vorhersagbar ist und damit optimal entworfen werden kann. Ein fehlerfreier Dialog sollte also sehr positiv bewertet werden, so die Annahme, und Abstriche in der Bewertung kommen durch Fehler bzw. die Reaktion des Systems auf diese zustande. Mglw. spielt auch der Kontext des Gebrauchs eine Rolle; so ist z. B. ein Smart-Home-System für ältere Menschen hilfreicher als für junge, da erstere evtl. nicht mehr mobil sind oder ein komplexes Gerät wie den Videorecorder nicht über dessen Bedienelemente zu programmieren in der Lage sind. In diesem Fall würde sicherlich das Urteil positiver ausfallen als bei einer Person, die nicht auf das System angewiesen ist. In dem beschriebenen Experiment war jedoch der Gebrauchskontext für die verschiedenen

Teilnehmer ziemlich gut vergleichbar, da die Gruppe relativ homogen war und die Situation für alle genau gleich. Damit sollten die Fehler doch stärker ins Gewicht fallen als sie es tun. Deshalb sollen in diesem Kapitel Gründe für das undeutliche Ergebnis gesucht werden.

Prinzipiell gibt es zwei mögliche Quellen für ein „unrealistisches“ Ergebnis: Das Testdesign oder die Analysemethode. Zum Testdesign ist generell anzumerken, dass es für diese Analyse in mancher Hinsicht sehr günstig war. So waren die Aufgaben in allen Dialogen gut vergleichbar und deckten einen relativ vielfältigen Bereich der Interaktionsmöglichkeiten mit dem System ab. Z. B. untersuchte die Lampenaufgabe die Schwierigkeit der Lokalisation von Gegenständen im Raum, die Fernseheraufgabe das Problem, dass dynamische Inhalte und Zeitpunkte kommuniziert werden mussten, und es mussten Objekte aus früheren Aufgaben in späteren Aufgaben wieder verwendet werden, wodurch das Verständnis der Benutzer für das (fehlende) „Gedächtnis“ offen gelegt wurde.

Zu den Szenariobeschreibungen, in denen die Versuchspersonen die Aufgaben präsentiert bekamen, muss jedoch angemerkt werden, dass sie in einigen Fällen bestimmte Arten von Fehlern suggerierten (dies ist letztlich auch in dem vorangehenden Absatz impliziert). Dadurch ist die Häufigkeit des Auftretens bestimmter Fehler mit einer bestimmten Konsequenz nicht nur von dem System und Benutzer abhängig, sondern wird von dem Test verfälscht. Ein Beispiel hierfür stellen Attributfehler dar, die in fast allen Dialogen bei der Aufgabe, die „Heute“-Nachrichten anzuschauen, auftraten, da hier durch die Verwendung des Namens in der Beschreibung die Verwendung des Namens auch bei der Interaktion mit dem System suggeriert wurde. Der Sendungsname „heute“ wurde vom System jeweils als Wert für das Attribut „Tag“ interpretiert, was jedoch in diesem Fall der richtige Zeitpunkt war. Die Konsequenz war also PARTIAL PROGRESS. Wäre die Sendung hingegen für „morgen“ zu bestimmen gewesen, hätte dies REGRESSION als Konsequenz gehabt. Auch andere vorkommende Attributfehler haben meistens eher negativere Konsequenzen, z. B. STAGNATION. Durch die Überdimensionierung des glücklich verlaufenen Sonderfalls wird also die Klasse zweifältig, mit anderen Worten, würde man die erstgenannten Fälle abspalten, wäre das Ergebnis der Prädiktion evtl. deutlicher.

Eine weitere Schwäche des Datensatz bzgl. der Prädiktion von Qualitätswahrnehmung aus Benutzerfehlern könnte darin liegen, dass das INSPIRE-System zum Zeitpunkt des analysierten Experiments noch relativ unausgereift war. Es gibt also (im Gegensatz zu dem oben Gesagten) einige Schwachpunkte des Systems, die von den Fehlern unabhängig existieren, möglicherweise aber das Urteil stärker beeinflussten als diese. Anlass zu dieser Vermutung gibt u. a. die Befragung der Teilnehmer am Ende des Versuchs. Hier wurden laut Boland et al. (2004) eher prinzipielle Aspekte des Systems bemängelt als die Fehlerhäufigkeit bzw. der Dialogfluss. Typische Beispiele für die Verbesserungswünsche der Probanden sind kürzere System-Prompts oder die Möglichkeit, zwei Anrufbeantworternachrichten gleichzeitig löschen zu können.

Während das Testdesign im Nachhinein nicht verändert werden kann, also eine unveränderliche Prämisse darstellt, kann eine Kritik des Analyseverfahrens Verbesserungen für weitere Untersuchungen anregen. Zuerst richtet sich hier das Augenmerk auf die Berechnungen, die mit den Daten durchgeführt wurden. Die Probleme der ungünstigen Verteilungen der Variablen wurden bereits geschildert und lassen sich nicht vermeiden, jedoch wäre die Signifikanzschwelle leichter überschreitbar gewesen, wäre nur die Signifikanz der Spearman-Korrelation ausgewertet worden. Dies hätte andererseits natürlich Einschränkungen zur Folge bzgl. der Vorhersageschärfe bzw. der Methoden, die für komplexere Vorhersagemodelle zur Verfügung stehen. Ob die Ergebnisse der Pearson-Korrelation zuverlässig genug sind, um die entsprechenden Möglichkeiten parametrischer Verfahren sinnvoll zu nutzen, ist allerdings nicht unstrittig.

Auch bezüglich der Reduktion der Paare, deren Korrelation untersucht wurde, wären andere Ansätze mglw. besser gewesen. So hätte man z. B. einige Fragen, mit denen ein Zusammenhang ad hoc als unwahrscheinlich eingestuft werden kann (z. B. Urteile, die die Systemstimme betreffen), aussortieren können. Oder man hätte die Urteile, für die eine Korrelation am plausibelsten erscheint, auswählen können. Mit den derart reduzierten Urteilen hätte man ggfs. wiederum ein Faktorenanalyse durchführen oder für jedes einzeln die Korrelation mit den Fehlerklassen berechnen können. Im Falle einer Faktorenanalyse wären die Faktoren evtl. sogar „schärfer“ gewesen als jene, die aus allen Urteilen

berechnet wurden, insofern als die Gesamtheit der verbliebenen Urteile schon konsistenter gewesen wäre.

Ein weiterer Verbesserungsvorschlag, der die Fehlerklassen betrifft, wäre, diese stärker auf die wahrgenommene Realität der Versuchspersonen zu münzen, denn schließlich spiegelt sich ja diese und nicht die objektive[21] in den Urteilen wider. Evidenz für die Diskrepanz zwischen wahrgenommener und objektiver Realität findet sich u. a. in den Ergebnissen der Befragung der Versuchspersonen am Ende des Tests. So ist die Wahrnehmung von Vokabelfehlern offensichtlich umfassender als bei der verwendeten Klassifikation, denn während einige Probanden ein zu kleines Vokabular bemängelten, äußerte keiner einen zu geringen Anteil an Verben oder Nomen, noch störte sich eine Testperson an einer zu geringen Anzahl grammatischer Varianten, die das System versteht. Das gegenteilige Problem besteht beim INADEQUATE-MAGNITUDE-OF-CONTROL-Fehler, der sowohl die falsche Einschätzung der Eigenschaften des Systems selbst (was sich z. B. in dem Versuch äußert, zwei Lampen gleichzeitig zu bedienen) als auch der bedienbaren Geräte (z. B. eine Lampe dimmen, die nur eine Helligkeitsstufe hat) erfasst. Der Unterschied besteht darin, dass die Steuermöglichkeiten der Lampe nicht bewertet werden sollten. Entsprechend wurde auch bei den Interviews als Mangel genannt, dass mit dem System nicht zwei Geräte gleichzeitig bedient werden können, nicht jedoch, dass man die rechte Stehlampe nicht dimmen kann o. ä.. Nicht ganz klar ist, ob Fehler, bei denen die Äußerung teilweise verstanden wird (PARTIAL PROGRESS), von den Benutzern als Fehler wahrgenommen werden oder nicht. Bei Ausschluss dieser würden sich die Fehlerzahlen deutlich verändern (36% der Fehler haben diese Konsequenz!). Bei einer besseren Abbildung der psychischen Realität der Versuchsperson bei der Bildung der Variable, die mit einem Urteil korreliert wird, könnte man eine stärkere Beziehung zwischen diesen erwarten.

[21] Die Realität, wie sie durch die Fehlerklassifikation erfasst wird, ist selbstverständlich nicht wirklich objektiv. Der Begriff ist hier eher als Antonym zu „subjektiv“ zu verstehen.

3.4.2 Angepasstes Fehlerklassifikationsschema

Auf Grundlage dieser Überlegungen wurde das Klassifikationsschema erneut modifiziert. Die Fassung, die für die Codierung der BoRIS-Daten genutzt wurde, soll hier wiedergegeben werden:

ERROR DEFINITION

- For each task there's a set of Attribute-Value-Pairs (AVPs) that has to be acquired and that is "hidden" from the user's immediate perception,
- execution of the task involves commanding the system and interpreting the cues in its response so as to transform the initial state of the system to the hidden goal state.
- "Optimal step" = maximum number of correct AVPs and no false AVPs (with respect to task given to the user) that can be deducted from the utterance are acquired by the system.
- In evidencing these deviations, the coding primarily deals with overt behaviour, not with the immediate and latent reasons that cause them (inferred reasons are however described in the notes-column of the sheet). In other words, the classification taps the "phenotypes" of errors (overt behaviour classifiable as an error) rather than their "genotypes" (causes of errors).
- the unit of analysis is one utterance (input);(the system prompt does not disqualify a progressive utterance)
- more than one error can be made in any one exchange…this will have to be refined still

ERROR TYPES

0. Technical Errors

- technical errors are coded for the purpose of accounting for otherwise unexplained interaction discrepancies only. Since the coding deals with human error, these errors will not be subject to successive considerations. Analysis of ASR errors will be done in conjunction with the logging parameters analysis.

ASR Error

- def= System does not understand due to ASR error

System/Wizard of Oz Error

- def= any irregularities that can be drawn back to wizard or system malfunction

1. Goal Level Errors

- def= Issuing a command for action that cannot be performed by the system because it does not possess that capability. It is possible to think of an extension to the system periphery that would be able to perform the intended action.

Inexistent Object of Control

- def= Attempting to control an object (device or content) that does not exist in the system. Example: Asking the system to open the window. (One can easily think that there was a module in an improved version of the system that could do this.)

Magnitude of Control for End Device

- def= Attempting to control an end device at a level not possible for the system. Example: Asking the system to dim a light that actually can be controlled only at the on/off level. (One can easily think that an improved system could do this.)

Control Conflict error

- def= Asking the system to do something which it cannot because of external restrictions. Example: Asking the system to show a film broadcasted in the evening now, the presentation time of which however is not controlled by the system but some external agent (TV company). (One can think that an improved system could retrieve the film on demand.)

2. Task Level Errors
(i.e., not understanding how to reach the goal in interaction with the system),

State Error
- def= Issuing a command that is valid and progressive (in regard with the goal expressed in the task given to the user in the experiment) in one state of the dialogue, but not in the current one.

Magnitude of System Control
- def= Issuing a command that exceeds the control capabilities of the system, e.g. combining more than one solution of the solution table in one utterance

Attribute Type Error
- def= Issuing a command using a wrong value class, e. g. string instead of Boolean/number

Unprogressive State Error
def= Issuing a command that only produces AVPs which are in the system memory already

Half-progressive State Error
- def= Issuing a command some of which's acquired AVPs are in the system memory already

Regressive Decision
- def= Issuing a command that is valid in the current state of the dialogue, but regressive (with regard to the goal expressed in the task given to the user in the experiment) e.g. a command that deletes one or more values that have been positively acquired by the system

3. Representational Level
(i. e., referring to knowledge of the world which is not shared with the system)

Domain modelling Error
- def= The user categorizes the domain (e. g. “TV Shows”) in another way than the system does. Example: “I would like to watch an action movie” (This fails because the system does not distinguish between (movie) topics but between program formats)
There are two special domains which are annotated separately:

Time Modelling Error
- def= The user refers to categorization of events in time that is not understandable by the system. Example: The user asks TV programme for the evening by asking the system to show programmes after 5pm. However, the system models time by categorizing it to morning, afternoon, evening, and night, and does thus not know the meaning of 5pm although it is in practice the same than "evening". (This should not be confused with the state error, wherein order errors are related to dialogue structure, not the world.)

Space Modelling Error
- def= The user refers to the space in a way that is not understandable by the system. Example: “Please turn on the lamp that is on the right from the table lamp” (This fails because the system does not have a model of the relative positions of the lamps.) (Note: This should not be confused with vocabulary error in the case the reference is made in just one word.

4. Command Level Errors

No Input
- def= Failing to issue a command during the response interval where the system expects it to be issued.

Vocabulary Nonunderstanding
- def= User uses vocabulary the system does not know

Vocabulary misunderstanding
- def= User uses vocabulary which is unintendedly understood by the system

Grammar Error
- def= User uses grammatical construction that is not understandable to the system although it knows the vocabulary

Reference
- def= Issuing a command that would demand of the system to remember former tasks/keep track of the dialogue history. It is possible to imagine another kind of awareness of the world, this error would not emerge.
E.g.:
- reference to dialogue history
The user refers to an earlier state of the scenario/dialogue the AVPs of which have been deleted since then. Example: “Please record the film I asked you to remind me of tonight”, “the table lamp as well, please” (after asking the system to switch on the standard lamp)
- indirect reference
The user defines a value by relating it to an earlier solution in the scenario, the AVPs of which have been deleted since then, or to the system periphery (devices) state. Example: “switch on the other lamp”, “Please dim the table lamp lower than the standard lamp”
- deictic expressions

IMMEDIATE CONSEQUENCES

1. Stagnation.
No AVPs are acquired with the utterance
- A special case of this is called Repetition. The system repeats the same prompt (word to word, complete or just a part of it).

- Another special case of this is called Help. The system plays the help prompt of the current GDN (in which the action alternatives are proposed to the user)

2. Regression.
The system goes to a state that is not on a direct path to the goal.

- A special case of this is called Restart. The system returns to its initial state, losing any progress achieved in the task before the error occurred.

3. Partial Progress.
The system goes to a state which is closer to the task goal, but not all the information in the user’s utterance is processed.

Zusätzlich zu diesen Klassen wurde in einer Extra-Spalte annotiert, wenn der Benutzer sein (Teil-) Ziel änderte, ohne dass die Situation dies erfordert hätte. Ein Beispiel für das INSPIRE-System wäre, dass der Benutzer zuerst die Tischlampe einschalten will, da dies aber vom System nicht sofort verstanden wird schließlich die linke Stehlampe benutzt.

4 Klassifikation der Fehler mit dem BoRIS-Restaurant-informations-System

4.1 Datenerhebung

4.1.1 Beschreibung des BoRIS-Systems

Um die Tauglichkeit des Klassifikationsschemas zur Beschreibung von Interaktionsproblemen für andere Systeme als INSPIRE zu testen, wurden Dialoge mit einem weiteren Sprachdialogsystem annotiert. BoRIS (Bochumer-Restaurant-Informations-System) ist ein SDS, mit dessen Hilfe ein Benutzer am Telefon ein Restaurant in der Bochumer Gegend suchen kann. Dazu sammelt das System in einem Dialog verschiedene Auswahlkriterien, die letztlich auf ein oder mehrere Restaurants verweisen, die dann vom System ausgegeben werden. Die Kriterien, nach denen ein Restaurant gesucht werden kann, sind die Art des Essens, der Ort des Restaurants, der Tag und die Tageszeit, für die das Essen geplant ist, sowie die Preiskategorie des Lokals. Das BoRIS-System wurde von Möller im Rahmen des „InfoVox"-Projektes entwickelt, wobei die Dialogstruktur der Ecole Polytechnique Fédérale de Lausanne (EPFL) entstammt. Das Konzept wurde vom Berkeley Restaurant Project (Jurafski et al. 1994) abgeleitet. Der erste Entwurf Möllers war für die schweizer Stadt Martigny bestimmt und in französischer Sprache, wurde später dann für die deutsche Sprache und den Raum Bochum adaptiert.

Wie das INSPIRE-System modelliert auch BoRIS die Aufgaben bzw. Restaurantlösungen mit Attribut-Wert-Paaren, die den o. g. Kriterien für die Restaurantsuche entsprechen. Auch das Dialogmanagement ist effektiv dem des INSPIRE-Systems sehr ähnlich, obwohl BoRIS nicht nach dem GDN-Prinzip funktioniert. Wie INSPIRE ist auch diesem System die Eigenschaft „mixed-initiative" zuzuschreiben. So spielt auch BoRIS zu Beginn des Dialogs einen offenen Prompt ab, in dem der Benutzer die Auswahlkriterien genannt bekommt und dann selbst die Kriterien wählen kann, nach denen er ein Restaurant suchen möchte. Zudem können auch hier mehrere AVPs in einer Äußerung spezifiziert werden, so dass wie bei INSPIRE nicht genau auf die vorangehende Systemfrage geantwortet werden muss.

Ein Unterschied zu INSPIRE besteht jedoch darin, dass es Attribute gibt, deren Spezifizierung lediglich der Dialogführung dient. Z. B. kann einer Äußerung die Information entnommen werden, welches Attribut der Benutzer spezifizieren möchte. So wählen einige Benutzer in dem offenen Start-GDN zuerst das Attribut (z. B. Ort des Restaurants), bevor sie in der nächsten Äußerung dessen Wert bestimmen. Weiterhin kann BoRIS mit verschiedenen „Confirmation Strategies“ konfiguriert werden:

- explizite Bestätigung jedes gefundenen AVPs
- implizite Bestätigung bei Fortschreiten des Dialogs oder
- zusammenfassende Bestätigung aller AVPs nach der Spezifikationsphase

Schließlich kann im Unterschied zu INSPIRE bei BoRIS nicht explizit nach Hilfe gefragt oder das System in den Ausgangszustand zurückgesetzt werden.

Die Sprachverarbeitung bei BoRIS ist der des INSPIRE-Systems sehr ähnlich. Beide Systeme können mit einem automatischen Spracherkenner (keyword spotting) betrieben werden, der durch einen Wizard-of-Oz ersetzt werden kann. Auch bei BoRIS wird der semantische Gehalt einer Äußerung ermittelt, indem die Äußerung nach relevanten Stichworten durchsucht wird, die auf kanonische Werte für die Attribute verweisen. Für die Sprachausgabe stehen entweder Aufnahmen realer Sprecher zur Verfügung, oder, und dies ist ein Unterschied zu INSPIRE, eine TTS-Sprachgenerierung (bzw. Aufnahmen einer TTS-Stimme). Im Fall von Confirmation-Prompts werden die Sätze mittels „Slot Filling“ generiert, d. h., die betreffenden Attributwerte, die bestätigt werden sollen, werden an entsprechender Stelle in den Satz eingefügt.

Die Informationen, die an den Benutzer ausgegeben werden, bezieht BoRIS aus einer Datenbank mit ca. 170 Restauranteinträgen. Diese kann lokal als Text-Datei oder über ein HTML-Interface abgefragt werden. Anhand der AVPs wird das passende Restaurant spezifiziert (pattern matching). Wird mit den gesammelten AVPs ein Restaurant gefunden, nennt BoRIS dessen Name und Adresse, wobei bis zu drei Restaurants in einem Prompt ausgegeben werden. Gibt es mehr passende Einträge, kann der Benutzer entscheiden, ob er diese hören möchte. Wie bei INSPIRE wird auch von diesem System der Dialog beendet, wenn durch die bisherigen AVPs eine ausreichend kleine Anzahl an Lösungen (bzw.

Restaurants) möglich ist (in diesem Falle drei Lösungen). Existiert kein passendes Restaurant, teilt das System dies mit. Der Benutzer kann daraufhin die Anfrage modifizieren, bekommt jedoch keine Information darüber, welche Modifikationen zu einem Ergebnis führen würden.

4.1.2 Versuchsbeschreibung

Die Versuche mit dem BoRIS-System, von denen die Transkriptionen und Benutzerurteile analysiert wurden, fanden wie das INSPIRE-Free-Wizard-of-Oz-Experiment am Institut für Kommunikationsakustik der Ruhr-Universität Bochum statt. An dem Test nahmen 40 Versuchspersonen teil, die zwischen 18 und 53 Jahre alt waren (Durchschnittsalter: 29 Jahre) und größtenteils mit der Stadt Bochum und in gewissem Maße den Restaurants dort vertraut waren. Jeder Teilnehmer hatte fünf Dialoge mit dem System zu führen, wobei die Kriterien der Restaurantauswahl im Vorhinein festgelegt wurden. Vier der Aufgaben waren durch die Versuchsleitung vorgegeben und für alle Probanden einheitlich, die fünfte konnte sich der Versuchsteilnehmer selber ausdenken, musste sie jedoch vor dem Dialog festlegen. Eine Liste der verwendeten Szenarios findet sich im Appendix (7.3). Manche der vorgegebenen Aufgaben ließen einige Attribute offen, und es wurde darauf geachtet, keinen bestimmten Wortlaut oder Vokabular zu suggerieren. Um dies zu vermeiden, wurden einige Kriterien ikonisch dargestellt; z. B. wurde der gewünschte Ort des Restaurants auf einer Landkarte markiert, oder die Preisklasse wurde durch eine Markierung auf einer Skala festgelegt. In manche Szenarios war zudem für den Fall, dass die Datenbank kein entsprechendes Restaurant enthält, ein Attribut vorgegeben, das für eine neue Anfrage modifiziert werden sollte. Von den insgesamt 200 Dialogen wurden drei wegen technischer Probleme aussortiert.

Auch bei diesem Test wurde die Wizard-of-Oz-Methode genutzt, jedoch wurden diesmal automatisch Fehler in der Transkription generiert, um den Einfluss der Fehlerrate (Word Error Rate) eines automatischen Spracherkenners auf den Dialogverlauf und die Qualitätsbeurteilung zu untersuchen. Dazu wurde von Skowronek (2002) ein Algorithmus entwickelt, der Fehler bei der automatischen Spracherkennung möglichst realistisch,

jedoch mit kontrollierbarer Fehlerrate, modelliert.[22] Dieser wird auf den Text angewandt, den der Wizard eingegeben hat, woraufhin der *verfälschte* Text an das Sprachverständnismodul weitergegeben wird und dort entsprechende Fehler produziert. In dem Experiment wurden mittels des Algorithmus Zielerkennungsraten von 60, 70, 80, 90 oder 100 Prozent für die Dialoge simuliert.

Weiterhin sollte in dem Versuch der Einfluss der Systemstimme (natürlich oder TTS) sowie der Confirmation-Strategie (implizit oder explizit) auf die Benutzerurteile und den Dialogverlauf getestet werden. Deshalb wurden zehn verschiedene Konfigurationsvarianten verwendet, für die die Stimme, WER und Confirmation-Strategie kombiniert wurden. Die Zuordnung der Systemkonfigurationsvarianten zu den unterschiedlichen Bedienungsszenarios erfolgte mit Hilfe eines Euler-Quadrates, so dass Störeffekte, die durch die Kombination hätten zustande kommen können, vermieden wurden.

4.1.3 Benutzerurteile

Wie beim INSPIRE-Experiment wurde zu Anfang und zu Ende des Experiments, sowie nach jeder Interaktion (Szenario) ein Fragebogen von den Versuchspersonen beantwortet. Die Fragen entstammen der Literatur und eigenen Überlegungen bzw. Untersuchungen Möllers zu den interessierenden Qualitätsaspekten. Mit Hilfe eines instruierenden Textes wurden die Teilnehmer auf die Interaktionen vorbereitet. Im Gegensatz zu dem beschriebenen INSPIRE-Versuch gab es jedoch keine Geschichte, die die Teilnehmer in eine reale Benutzungssituation versetzte, sondern sie wurden nur gebeten, sich eine solche Situation vorzustellen. Die Gespräche zwischen Benutzer und System verliefen über ein Telefon, wobei die Übertragung simuliert wurde, um gleiche Bedingungen bei allen Dialogen zu gewährleisten.

Der erste der drei Fragebögen (A) wurde bei den folgenden Korrelationsberechnungen nicht genutzt, weil hier keine Qualitätsurteile zu dem System abgegeben wurden. Auch der Fragebogen zu Ende des Tests (C) wurde – wie bei der Analyse der INSPIRE-Daten –

[22] Wiedergegeben in Möller (2005b)

nicht ausgewertet. Der zweite Fragebogen (B), der nach jeder Interaktion von der Versuchsperson ausgefüllt wurde, ist wie beim INSPIRE-Experiment in Sektionen gegliedert, die jeweils bestimmte Aspekte der Interaktion betreffen. Diese Aspekte sind auch weitgehend kongruent mit denen des INSPIRE-Fragebogens. Die Fragen selbst unterscheiden sich jedoch sowohl bezüglich der abgefragten Eigenschaft als auch bezüglich der Art der Fragestellung. Während beim INSPIRE-Fragebogen Aussagen formuliert wurden, auf die immer mit der gleichen Zustimmungsskala geantwortet wurde, wechseln hier die Skalentypen sowie die Attribute, mit denen die Skalenstufen benannt sind. Bei manchen der kontinuierlichen Skalen ist jede Stufe beschriftet, wohingegen für andere nur Antonyme an den Enden der Skala angebracht sind. Auch hier sind aber die Fragen als Aussagen formuliert. Der vollständige Fragebogen ist im Appendix (7.4) wiedergegeben.

Alle Skalen gingen über die fünf Stufen hinaus, ermöglichten also extreme Bewertungen, die teilweise auch gelabelt waren. Bei der Übertragung in die SPSS-Datei wurde jeweils das positivste Urteil als 5 eingetragen und das negativste mit 1. Für extreme Urteile an den Randbereichen der Skala blieben demnach die Werte 0 bzw. 6.

4.2 Korrelationsberechnungen

4.2.1 Vorbereitung der Daten

Trotz der Kritik, die in Kapitel 3.4.1 an dem Verfahren zur Bestimmung der Korrelationen zwischen den Fehler- und Konsequenzklassen und den Urteilen geübt wurde, sollen bei der Untersuchung der BoRIS-Daten die selben Koeffizienten und das selbe Signifikanzkriterium genutzt werden, um die Vorteile, die der Pearson-Koeffizient bietet, nutzen zu können und mit den Ergebnissen aus den INSPIRE-Daten vergleichbare Zahlen zu bekommen. Entsprechend wurden auch die 25 Urteile des Interaktionsfragebogens B wie beim INSPIRE-Versuch zu Faktoren zusammengefasst. Können die Ergebnisse verglichen werden, so ergeben sich daraus mglw. Rückschlüsse auf die Validität der Ergebnisse sowie Einflüsse anderer Aspekte der Interaktion auf die Bewertung. Zum Beispiel könnte bei einem wesentlich deutlicheren Ergebnis für die BoRIS-Daten angenommen werden, dass der Einfluss der Fehler auf die Urteile von anderen Einflüssen

überdeckt wurde. Sind die Ergebnisse sehr ähnlich, d. h., bestehen Korrelationen zwischen ähnlichen Fehlern und Urteilen, würde das die Validität der Untersuchung enorm stützen.

Da also wiederum Pearson-Korrelationen berechnet werden sollen, müssen auch die Fehlerdaten auf ihre Verteilung hin analysiert und, wenn dies die Verteilungseigenschaften verbessert, transformiert werden. Vorher können jedoch die seltenen Fehler, für die aufgrund ihres seltenen Auftretens weder die Verteilungstests noch die Berechnung von Korrelationskoeffizienten sinnvoll ist, aussortiert werden. Sie sollen später anhand der Streudiagramme analysiert werden. Als Schwellenwert wurden neun Instanzen der Fehler gewählt, da der nächst-seltene Fehler bereits 15 mal auftrat und somit zwischen diesen Häufigkeiten sinnvoll eine Grenze gezogen werden konnte. Von den Berechnungen ausgeschlossen wurden dementsprechend die Klassen

- NO INPUT (1)
- REFERENCE (0)
- INEXISTENT OBJECT OF CONTROL(2)
- MAGNITUDE OF DEVICE CONTROL (5)
- MAGNITUDE OF SYSTEM CONTROL (8)
- UNPROGRESSIVE STATE (8)
- REGRESSIVE DECISION (7)
- SPACE MODELLING (4)
- OTHER (4) und
- die Konsequenz HELP (4)

Außerdem wurden die Technischen Fehler nicht analysiert, da diese auch in den aus den Log-Dateien extrahierten Parametern erfasst werden und somit schon hinsichtlich der hier interessanten Aspekte analysiert wurden.

Alle anderen Fehler traten 15 mal oder häufiger auf, sind jedoch erwartungsgemäß nicht normalverteilt (Kolmogorov-Smirnov-Test). Ein K.-S.-Test auf Poissonverteilung bestätigt für die verbleibenden Fehlerklassen (hier sind die Konsequenzen nicht eingeschlossen!) die Vermutung, es handele sich um diese, für Fehlerhäufigkeiten typische Verteilung, da alle (exakten) Signifikanzwerte größer als 0,5 sind, außer für den DOMAIN-MODELLING-Fehler für den die (exakte) Signifikanz mit 0,32 aber auch deutlich über dem

gewählten Niveau von 0,05 liegt. Deshalb sollen von diesen Variablen die Quadratwurzeln gezogen werden.

Bei den Konsequenzklassen ergibt der Signifikanztest für RESTART 0,168 und für PARTIAL PROGRESS 0,220 (beide exakt). Da RESTART maximal einmal pro Dialog auftauchen kann – als RESTART wurde ein Abbruch durch das System kodiert – hat das Wurzelziehen in diesem Fall keine Auswirkung auf die Verteilungsfunktion. Hingegen zeigt für die Konsequenz PARTIAL PROGRESS, die im Test einen (exakten) Signifikanzwert von 0,22 aufweist, das Histogramm eine Poisson-ähnliche Verteilung, weswegen hier die Wurzelwerte verwendet werden sollen. Die Konsequenzklassen STAGNATION, REPETITION und REGRESSION folgen dem Testergebnis zufolge auch der Poissonverteilung nicht, zeigen aber im Histogramm eine deutliche Tendenz zu dieser. Zudem gibt es in allen jeweils 1-2 Ausreißer. Dieses Problem soll mit der Wurzeltransformation zu lindern versucht werden.

4.2.2 Faktorenanalyse der Benutzerurteile

Auch zu diesem Experiment lag bereits eine Faktorenanalyse von Möller (2005b) vor, in der die Fragen des Interaktionsfragebogens in fünf perzeptive Dimensionen gruppiert wurden. Diese können nach Möller wie folgt interpretiert werden:

- *Factor 1: Overall impression of the system, informativeness, overall behaviour of the system, interaction capability/flexibility, user satisfaction. Related questions: B0, B1, B2, B3, B4, B9, B10, B13, B14, B23.*
- *Factor 2: Overall impression of the system, intelligibility, friendliness, naturalness, pleasantness. Related questions: B0*[23]*, B7, B16, B18, B22, B24.*
- *Factor 3: Cognitive effort and dialogue smoothness. Related questions: B6, B8, B19, B21.*
- *Factor 4: Speed and conciseness. Related questions: B15, B17, B20.*
- *Factor 5: Reaction like a human. Related question: B12.*[24]

[23] Beim Nachrechnen der Faktorenanalyse entsprechend den Angaben Möllers ergab sich, dass B0 tatsächlich nur auf den ersten Faktor lädt. In Möllers Ladungstabelle finden sich für beide Faktoren exakt die selben Ladungen für dieses Urteil, es handelt sich also offenbar um einen Copy-and-Paste-Fehler bei der Erstellung der Tabelle. Die übrigen Werte wurden exakt so gefunden wie angegeben. Im Folgenden soll deshalb B0 nicht als Komponente von Faktor 2 berücksichtigt werden.

[24] Möller 2005b, S. 261 f.

Hier der Schlüssel für die Kürzel der Fragen:

Faktor 1:

- B0: Gesamteindruck
- B1: Das System lieferte die gewünschten Auskünfte (trifft zu – trifft nicht zu)
- B2: Die gelieferten Auskünfte waren… (vollständig – unvollständig)
- B3: Die Auskünfte waren…(klar – unklar)
- B4: Sie schätzen die Auskünfte ein als (falsch – richtig)
- B9: Ihrer Einschätzung nach verarbeitete das System ihre Angaben richtig (trifft (nicht) zu)
- B10: Das System reagierte immer wie erwartet (trifft (nicht) zu)
- B13: Das System reagierte (flexibel – unflexibel)
- B14: Sie konnten das Gespräch wie gewünscht lenken (trifft (nicht) zu)
- B23: Sie sind mit dem Gespräch insgesamt zufrieden

Faktor 2:

- B7: Wie gut war das System akustisch verstehbar? ((extrem schlecht) – schlecht – dürftig – ordentlich – gut – ausgezeichnet – (ideal))
- B16: das System reagierte (freundlich – unfreundlich)
- B18: Sie empfanden das Gespräch als (natürlich – unnatürlich)
- B22: Die Stimme des Systems war (natürlich – unnatürlich)
- B24: Sie empfanden das Gespräch als (angenehm – unangenehm)

Faktor 3:

- B6: Sie mussten sich sehr konzentrieren, um zu verstehen, was das System von ihnen verlangte (trifft (nicht) zu)
- B8: Sie wussten an jeder Stelle, was das System von ihnen verlangte (trifft (nicht) zu)
- B19: Das Gespräch verlief… (übersichtlich - verwirrend)
- B21: Das Gespräch verlief… (glatt – holprig)

Faktor 4:

- B15: Das System reagierte… (zu schnell – angemessen – zu langsam)
- B17: Die Äußerungen des Systems waren… (kurz – lang)
- B20: Das Gespräch war… (zu kurz – angemessen – zu lang)

Faktor 5:

- Das System reagierte wie ein Mensch (trifft (nicht) zu)

Nach einer genaueren Analyse können die Faktoren 1 und 2 auch etwas griffiger umschrieben werden. Der erste Faktor repräsentiert Urteile über den Gesamteindruck und die Zufriedenheit des Benutzers, ferner (bzgl. der System-Prompts) Klarheit, Richtigkeit, Vollständigkeit und Wunschgemäßheit der gegebenen Auskünfte, sowie (bzgl. der Verarbeitung der Eingaben) generell die Verarbeitung der Eingaben des Benutzers, sowie im Speziellen die Erwartungsmäßigkeit und Flexibilität der Reaktion des Systems und die Lenkbarkeit des Dialogs. Damit ließe sich dieser Faktor mit der Wahrgenommenen Qualität insgesamt und speziell der Qualität des Informationsaustausches (Quality of Information Exchange) bezeichnen. Die Zugehörigkeit von Gesamteindruck und Zufriedenheit zu dieser Dimension deutet im Übrigen darauf hin, dass die Qualität des Informationsaustauschs für die Gesamtqualität entscheidend ist, womit die Bedeutsamkeit von Interaktionsfehlern für die Vorhersage von Qualitätsurteilen bestätigt wird[25].

Der zweite Faktor umfasst Urteile zur Verständlichkeit und Natürlichkeit der Stimme des Systems, dessen Freundlichkeit sowie die Natürlichkeit und Angenehmheit des Gesprächs. In Übereinstimmung mit den in Möller (2005) gebrauchten Begriffen und Konzepten bei der Benennung von Urteilsdimensionen könnte man deshalb sagen, dieser Faktor beschreibe die Persönlichkeit (Personality) des Systems.

[25] Es sei jedoch nochmals angemerkt, dass es sich bei den hier erfassten Fehlerklassen in gewissem Sinne um objektive Klassen handelt, d. h., Defizite im Informationsaustausch werden vom Nutzer mglw. anders wahrgenommen und gewichtet. Deshalb wird diese Untersuchung durch die o. g. Erkenntnis nicht überflüssig.

Da auch die Faktorenanalyse ein parametrisches Verfahren ist, das u. a. Normalverteilung und Intervallskalenniveau voraussetzt, soll an dieser Stelle darauf hingewiesen werden, dass der Datensatz den Ansprüchen des Verfahrens streng genommen nicht genügt. Während die kontinuierlichen Bewertungsskalen des Fragebogens immerhin eher die Annahme des Intervallskalaniveaus zulassen als die bei INSPIRE (hauptsächlich) verwendete fünfstufige Skala, sind die Verteilungen der Urteile Möller (2005b) zufolge nicht normal. Damit würden die gefundenen Dimensionen eigentlich in Frage gestellt, was die folgende Korrelationsanalyse müßig machen würde, jedoch stimmt der Erfolg bei der Benennung der Faktoren zuversichtlich, dass das Ergebnis als valide betrachtet werden kann.

4.2.3 Korrelationen mit den Fehlerklassen

Obwohl die Analyse der Verteilungen der Urteile darauf hindeutet, dass auch bei diesem Versuch eine Teilung des Datensatzes nach Systemkonfiguration zu genaueren Ergebnissen führen würde (die Verteilungen sind „normaler“, wenn die Daten nach Konfigurationen getrennt wurden), soll hier der vollständige Datensatz zur Berechnung der Korrelationen verwendet werden. Dies wird damit begründet, dass in dem Experiment sehr viele Konfigurationsvarianten getestet wurden, so dass der Datensatz bei einer Teilung nach diesem Kriterium erheblich schrumpfen würde. Hinzu kommt, dass die fünf Szenarien sehr unterschiedlich schwer zu lösen waren[26], d.h., diese könnten einen erheblichen Einfluss auf die Bewertung haben. Schließlich spricht gegen eine Teilung der Daten, dass insgesamt nur relativ wenige Fehler begangen wurden, so dass bei Verringerung der Fälle der Untersuchungsgegenstand gewissermaßen zerrinnt. Von einer Unterteilung der Daten nach anderen Variablen, z. B. der Erkennungsrate, wird abgesehen, da die Varianten nicht gleichmäßig über die Benutzer verteilt sind, d. h., einige Benutzer wären mit mehreren Dialogen in einem Sub-Datensatz vertreten, während andere Benutzer gar nicht darin auftauchen.

[26] Z. B. bedurfte es für die Aufgabe, Ente essen zu gehen, gewöhnlich nur einer Benutzeräußerung, während für die Aufgabe, in Grumme griechisch essen zu gehen, kein Restauranteintrag in der Datenbank existiert.

So wurden also Korrelationen zwischen den fünf genannten Faktoren und den verbleibenden, ggfs. transformierten Fehlerzahlen berechnet, wobei wiederum Signifikanz ($p<0,05$) für Pearsons *r* und Spearmans ρ als Kriterium der Bedeutsamkeit der Ergebnisse gewählt wurde. Aus den 70 Paarungen ergeben sich nach dem derart festgelegten Kriterium 20 signifikante Korrelationen (28%). Prozentual sind das doppelt so viele Zusammenhänge wie für das INSPIRE-System gefunden wurden. Zu diesem Vergleich muss gesagt werden, dass aufgrund der wesentlich höheren Anzahl der Fälle das Signifikanzniveau leichter überschritten wird (vgl. Bortz 1977).

Korrelationen BoRIS		
Faktor 1 (Quality of Information Exchange)	***R***	**ρ**
External Restrictions	-0,276	-0,258
Attribute Type	-0,222	-0,217
Time Modelling	-0,244	-0,230
Stagnation	-0,219	-0,218
Repetition	-0,186	-0,176
Regression	-0,486	-0,498
Restart	-0,478	- 0,435
Faktor 2 (Personality)	***R***	**ρ**
Partial progress	0,198	0,211
Attribute Type	0,163	0,179
Faktor 3 (Cognitive Effort)	***R***	**ρ**
Vocabulary Nonunderstanding	-0,152	-0,155
Stagnation	-0,347	-0,362
Repetition	-0,290	-0,276
Regression	-0,162	-0,162 [sic!]
Faktor 4 (Speed and Conciseness)	***R***	**ρ**
Vocabulary Nonunderstanding	-0,204	-0,167
Control Conflict (because of External Restrictions)	-0,330	-0,321
Stagnation	-0,338	-0,335
Repetition	-0,315	-0,300
Regression	-0,273	-0,230
Faktor 5 (Reaction Like A Human)	***R***	**ρ**
Stagnation	-0,183	-0,154
Repetition	-0,161	-0,124

Tabelle 4.1Signifikante Korrelationen zwischen Fehlerklassen und Faktoren der Benutzerurteile aus dem Experiment mit dem BoRIS-System.

Andererseits sind die Werte der Koeffizienten relativ niedrig ($0{,}127 < |r| < 0{,}486$; $0{,}124 < |\rho| < 0{,}498$).[27] Positiv fällt jedoch auf, dass nur zwei der signifikanten Korrelationen

[27] Es scheint bzgl. der Werte einen Zusammenhang mit der Anzahl der Fälle zu geben, da auch beim INSPIRE-System die Werte größer wurden, nachdem die Fälle geteilt worden waren.

größer null sind, d. h., alle anderen folgen der erwarteten Tendenz. Die positiven Korrelationen bestehen zwischen dem Faktor PERSONALITY und PARTIAL PROGRESS (r= 0,198; ρ= 0,211) sowie ATTRIBUTE TYPE (r= 0,163; ρ= 0,179). Eine Beziehung zwischen der Bewertung der Persönlichkeit des Systems (in Form seiner Stimme etc.) wäre nicht zu erwarten gewesen und zeigt sich auch nur in einer relativ schwachen Korrelation. Im Fall von PARTIAL PROGRESS ließe sich ein positiver Zusammenhang aber auch damit erklären, dass der Benutzer sich an Fehlern mit dieser Konsequenz nicht so sehr stört, da das System immerhin einen Teil der Äußerung verstanden hat (und die Aufmerksamkeit des Benutzers in seiner Antwort auf diesen fokussiert) und der Benutzer auch häufig nicht gemerkt haben mag, dass ein Teil der Information nicht verstanden wurde (weil er nicht weiß, dass das System tatsächlich alle AVPs wiederholt, die es gesammelt hat).

Von den anderen vier Faktoren korreliert besonders häufig der erste (Quality of Information Exchange) mit den Fehler- und Konsequenzzahlen.

- EXTERNAL RESTRICTIONS (r= -0,276; ρ= -0,258)
- ATTRIBUTE TYPE (r= -0,222; ρ= -0,217)
- TIME MODELLING (r= -0,244; ρ= -0,230)
- STAGNATION (r= -0,219; ρ= -0,218)
- REPETITION (r= -0,186; ρ= -0,176)
- REGRESSION (r= -0,486; ρ= -0,498)
- RESTART (r= -0,478; ρ= - 0,435)

Es wurde bereits angedeutet, dass mit diesem Faktor die besten Korrelationen erwartet wurden, da hier Urteile Einfließen, die, wie die Fehler, die Verständigung mit dem System betreffen. Insofern stellt allein die Tatsache, dass hier die meisten Korrelationen gefunden wurden, bereits ein positives Ergebnis dar. Die Konsequenzen REGRESSION und RESTART zeigen die höchsten gefundenen Koeffizientenwerte für r und für ρ, für EXTERNAL RESTRICTIONS, TIME, STAGNATION und ATTRIBUTE TYPE wurden moderate Werte gefunden.

Der EXTERNAL-RESTRICTIONS-Fehler ist, das muss hier erwähnt werden, in gewissem Sinne nicht als Interaktionsfehler zu betrachten, da das System hier in der Regel die vom Benutzer gegebene Information korrekt verarbeitet, jedoch kein Restaurant in der

Datenbank findet. Dieser Fehler beruht nicht auf einer Misskonzeption des Benutzers, denn würde dieser bereits die Einträge in der Datenbank kennen, bräuchte er BoRIS nicht zu benutzen. Einige dieser „Fehler“ sind sogar vom Szenario vorgegeben. Dennoch ist ein Einfluss auf das Qualitätsurteil sehr plausibel, da die entsprechende Antwort des Systems, wie gesagt, offen lässt, wie das Problem behoben werden könnte. Lediglich bei der Interpretation des Zusammenhangs ist Bedachtsamkeit geboten. So wurde der Fehler, wenn er wie oben beschrieben auftauchte, ohne Konsequenz annotiert, da der Dialogablauf dadurch nicht gestört wird.

Bei den Konsequenzen ist auffällig, dass im Gegensatz zum INSPIRE-System (sowie den anderen Wahrnehmungsdimensionen dieses Versuchs) REGRESSION (incl. RESTART) einen stärkeren Zusammenhang aufweisen als STAGNATION (inkl. REPETITION). Da das hier gefundene Verhältnis allein von der Begrifflichkeit her in beiden Fällen erwartet worden wäre, ist auch dieses Ergebnis positiv zu bewerten. Möglicherweise kann es auf die jetzt etwas intuitiver formulierte Definition der REGRESSION-Klasse zurückgeführt werden (obwohl der De-facto-Unterschied nicht sehr groß sein dürfte).

Der dritte Faktor, COGNITIVE EFFORT, korreliert mit den Fehler- bzw. Konsequenzklassen

- VOCABULARY NONUNDERSTANDING (r= -0,152; ρ= -0,155)
- STAGNATION (r= -0,347; ρ= -0,362),
- REPETITION (r= -0,290; ρ= -0,276) und
- REGRESSION (r= -0,162; ρ= -0,162 [sic!])

Die Korrelationen mit STAGNATION und REPETITION, die als moderat bis hoch eingestuft werden können, passen sehr schön zu allen in diesem Faktor enthaltenen Urteilen. So ist, wenn der Benutzer nicht versteht, was das System von ihm erwartet, Stagnation die logische Konsequenz. Die Korrelationen mit VOCABULARY NONUNDERSTANDING und REGRESSION sind sehr niedrig, jedoch nicht unplausibel. Dass hier REGRESSION schwächer korreliert als STAGNATION widerspricht dem im Zusammenhang mit Faktor 1 Gesagten nicht, da es sich hier um den Zusammenhang mit relativ *speziellen* Urteilen handelt. So ist z. B. für die Frage „Sie wussten an jeder Stelle, was das System von Ihnen verlangte“ (B8) ein Zusammenhang mit STAGNATION intuitiv plausibler als mit REGRESSION, während das

Verhältnis für die eher generellen Fragen, die Faktor 1 konstituieren (z. B. „Gesamteindruck“), umgekehrt ist.

Von den Urteilen, die Faktor 4, SPEED AND CONCISENESS, prägen, ist eigentlich nur für B20 (Das Gespräch war zu lang/kurz) eine Korrelation mit Fehlern denkbar, da sich weder die Länge der Prompts noch die Reaktionszeit des Systems auf das Benutzerverhalten auswirken sollten. Dennoch gibt es hier fünf signifikant korrelierende Paare, die allesamt moderate Werte für beide Koeffizienten aufweisen:

- VOCABULARY NONUNDERSTANDING (r= -0,204; ρ= -0,167)
- CONTROL CONFLICT (BECAUSE OF EXTERNAL RESTRICTIONS) (r= -0,330; ρ= -0,321)
- STAGNATION (r= -0,338; ρ= -0,335)
- REPETITION (r= -0,315; ρ= -0,300)
- REGRESSION (r= -0,273; ρ= -0,230)

Die Korrelationen der Fehler und Konsequenzen mit der Dialoglänge erklären sich eigentlich von selbst, so bedeuten STAGNATION und REGRESSION eine unnötige Verlängerung des Dialogs, wobei interessant, und ein plausibles Ergebnis ist, dass PARTIAL PROGRESS hier nicht korreliert. Für den EXTERNAL-RESTRICTIONS-Fehler gilt, dass die Angabe einer AVP-Konstellation ohne passendes Restaurant das Gespräch zwar nicht direkt unnötigerweise verlängert, aber dennoch wird bei häufigem Auftreten des Fehlers der Dialog sehr lang. Auch das Nichtverstehen von Vokabular führt in der Regel dazu, dass der Dialog sich verlängert, da bestimmte Informationen nochmals geäußert werden müssen. Dennoch ist es in diesem Zusammenhang erstaunlich, dass das Missverstehen von Wörtern (das definitionsgemäß REGRESSION, also die vermeintlich schlimmere Konsequenz, verursacht), keine Beziehung zu diesem Faktor aufweist.

Faktor 5, REACTION LIKE A HUMAN, hat zwei signifikante Korrelation: mit STAGNATION (r= -0,183; ρ= -0,154) und REPETITION (r= -0,161; ρ= -0,124). Auch hier waren Korrelationen nicht unbedingt zu erwarten, jedoch machen sie Sinn, da insbesondere wortgenaue Wiederholungen der selben Frage nicht menschlichen Kommunikationsformen entsprechen. Die Werte der Koeffizienten sind jedoch auch sehr niedrig.

4.2.4 Analyse der seltenen Fehler

Wie bereits angekündigt, sollen in diesem Abschnitt die Fehler- bzw. Konsequenzklassen analysiert werden, die nur sehr selten annotiert wurden. Wie bei der entsprechenden Analyse für die INSPIRE-Daten wird hier die Möglichkeit überprüft, dass ein Fehler bzw. eine Konsequenz zwar selten auftrat, jedoch schwere Auswirkungen auf die Bewertung des Systems hatte. Zwei Fehlerklassen können sofort auch von dieser Analyse ausgeschlossen werden: Der REFERENCE-Fehler trat in dem gesamten Datenkorpus nicht auf, und der NO-INPUT-Fehler wurde nur einmal annotiert, wobei sich nach der Kodierung herausstellte, dass hier die entsprechende Äußerung mitsamt dem restlichen Dialog verloren gegangen ist. Es handelt sich also hier nicht um einen Fehler, sondern ein Problem mit der Log-Datei.

Die Streudiagramme der verbleibenden Fehler jeweils über die fünf Faktoren zeigen insgesamt wenig Anzeichen von Zusammenhängen. Um mit dem INEXISTENT-OBJECT-OF-CONTROL-Fehler zu beginnen, der zwei mal vorkam (ein Fall wurde jedoch bei der Berechnung der Faktoren ausgeschlossen, da hier zwei Urteile fehlten): Dieser zeigt keine extremen, im Gegenteil sogar ziemlich durchschnittliche Werte aller Faktoren für den entsprechenden Dialog.

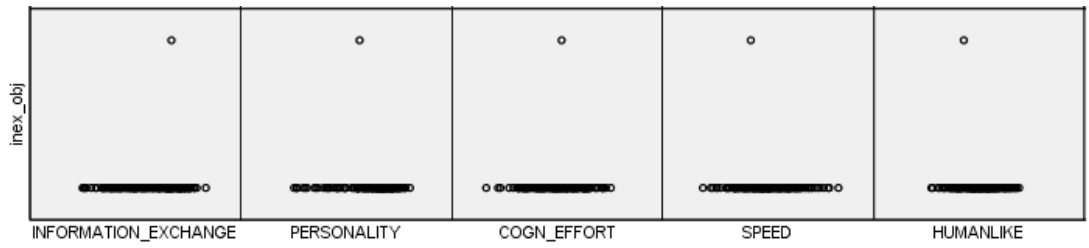


Diagramm 4.1 Streudiagramme des INEXISTENT-OBJECT-OF-CONTROL-Fehlers über den fünf Faktoren aus Möller (2005 b)

MAGNITUDE OF DEVICE CONTROL wurde fünf mal annotiert, wobei zwei Instanzen auf den selben Dialog fallen und eine wegen fehlender Faktorwerte nicht in den Streudiagrammen dargestellt ist. Eine schwache positive Beziehung der verbleibenden Fehler könnte man aus den Diagrammen mit den Faktoren PERSONALITY und COGNITIVE EFFORT herauslesen, wobei eine positive Beziehung an sich schon fragwürdig ist, insbesondere aber mit der

Persönlichkeit des Systems. Dieses Bild sollte also nicht überbewertet werden, da es wohl eher auf einer zufälligen Konstellation der Daten beruht.

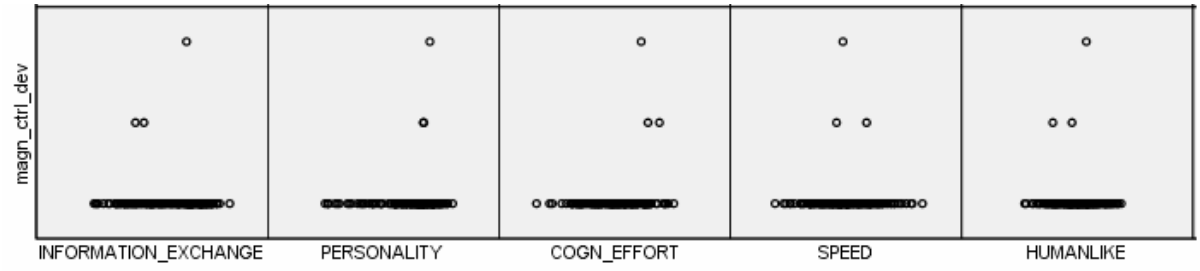


Diagramm 4.2 Streudiagramme des MAGNITUDE-OF-DEVICE-CONTROL-Fehlers über den fünf Faktoren aus Möller (2005b)

Für MAGNITUDE OF SYSTEM CONTROL deutet sich eine negative Korrelation zu Information Exchange sowie SPEED AND CONCISENESS an. Die Beziehung zur Geschwindigkeit und Flüssigkeit der Interaktion erklärt sich möglicherweise dadurch, dass dieser Fehler teilweise bedeutete, dass der Benutzer eine besonders komplexe Aufgabe mit der Äußerung bewirken wollte (z. B. drei Auswahlkriterien gleichzeitig fokussieren) oder auch versuchte, einem fehlgeschlagenen Kommando entgegenzuwirken (durch Zurücksetzen des Systems auf den vorherigen Zustand), was die schnellere Variante gewesen wäre, jedoch nicht funktionierte. Diese Fehler bedeuten letztlich auch, dass der Informationsaustausch nicht optimal funktionierte. Nicht erklärt werden kann die relativ deutliche positive Beziehung zu PERSONALITY, die sich erstaunlicherweise durch Korrelationen mit allen Fehlern zieht. Dennoch deutet sich an, dass dieser Fehler bei der Bewertung des BoRIS-Systems eine Rolle spielte.

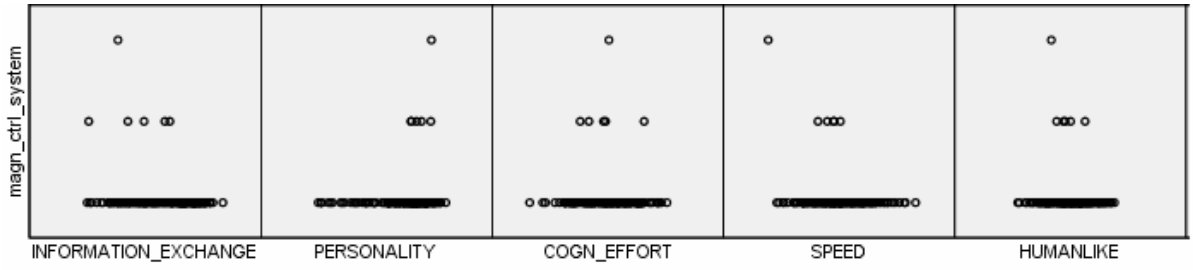


Diagramm 4.3 Streudiagramme des MAGNITUDE-OF-SYSTEM-CONTROL-Fehlers über den fünf Faktoren aus Möller (2005b)

Die nächsten Diagramme bilden das Verhältnis des UNPROGRESSIVE-STATE-Fehlers zu den fünf Faktoren ab. Der Fehler wurde acht mal annotiert, wobei zu einigen Dialogen,

wiederum wegen fehlender Urteile, keine Faktorwerte berechnet wurden. Ein Zusammengehen der Veränderung der Fehlerzahl mit den Faktorwerten lässt sich für keinen der fünf Faktoren ausmachen.

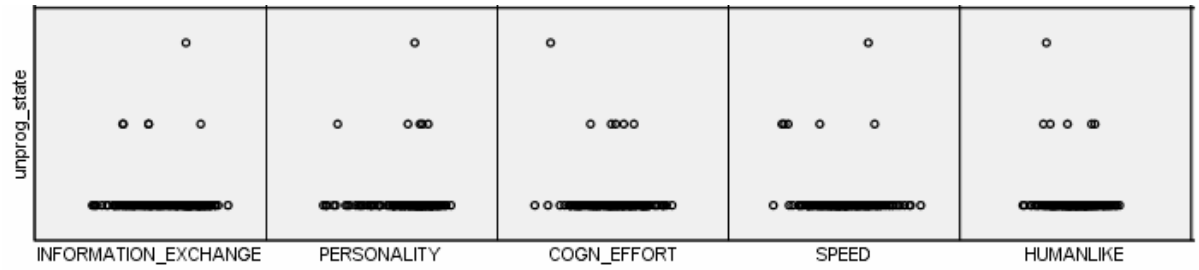


Diagramm 4.4 Streudiagramme des UNPROGRESSIVE-STATE-Fehlers über den fünf Faktoren aus Möller (2005b)

Auch beim REGRESSIVE-DECISION-Fehler, der sieben mal auftrat (ein Fall ist wieder nicht dargestellt), liegen die fehlerbehafteten Dialoge insgesamt mittig im Bewertungsbereich der fehlerfreien Interaktionen.

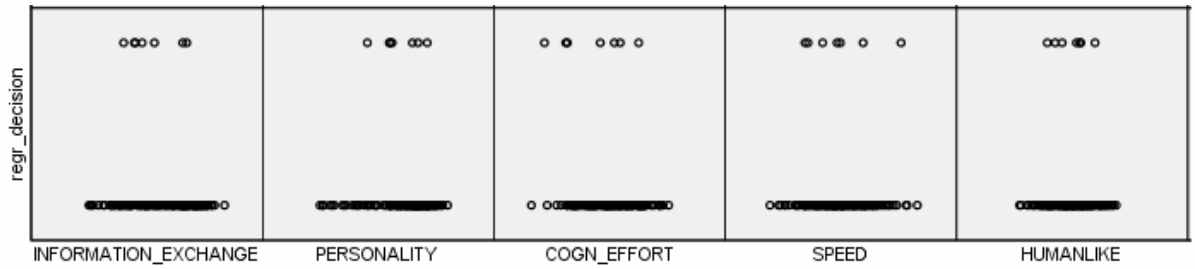


Diagramm 4.5 Streudiagramme des REGRESSIVE-DECISION-Fehlers über den fünf Faktoren aus Möller (2005b)

Das selbe gilt für den SPACE-MODELLING-Fehler sowie die Klasse sonstiger Fehler (OTHER), die jeweils vier mal annotiert wurden.

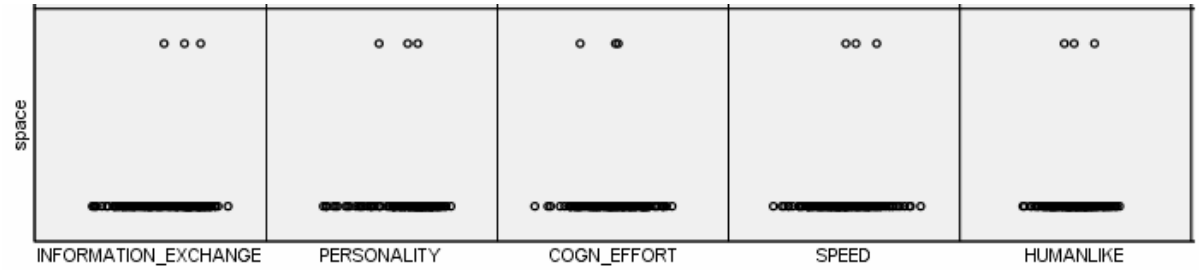


Diagramm 4.6 Streudiagramme des SPACE-MODELLING-Fehlers über den fünf Faktoren aus Möller (2005b)

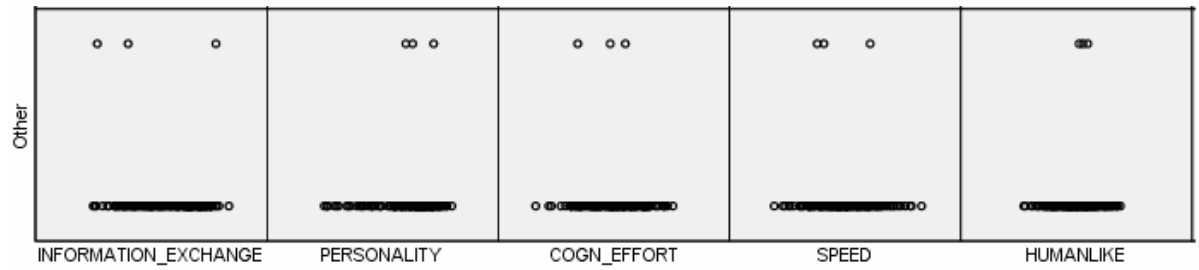


Diagramm 4.7 Streudiagramme der sonstigen Fehler über den fünf Faktoren aus Möller (2005b)

Schließlich die Konsequenz HELP. Sie wurde vier mal annotiert, wovon für einen Fall keine Faktorwerte berechnet wurden. Interessanterweise deuten die Streudiagramme nicht nur eine negative Beziehung mit COGNITIVE EFFORT an (hier ist der Kausalzusammenhang wohl entgegengesetzt zu den anderen Paaren zu sehen, insofern im Fall zu hoher kognitiver Belastung Hilfe-Prompts abgespielt werden sollten) sondern auch eine positive Korrelation mit dem Faktor Persönlichkeit des Systems, in den Urteile wie „Das Gespräch verlief angenehm“ einflossen. Die Neigung des PERSONALITY-Faktors, positiv mit den Fehlerklassen zu korrelieren, wurde bereits erwähnt. Eine sinnvolle Begründung ist auch hier schwierig.

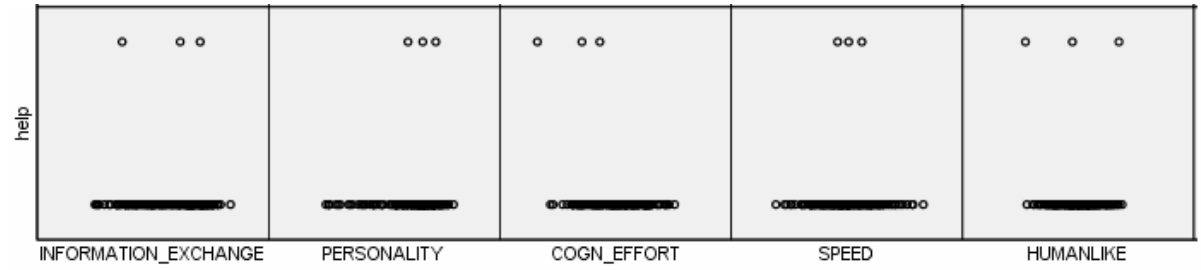


Diagramm 4.8 Streudiagramme der Konsequenz HELP über den fünf Faktoren aus Möller (2005b)

4.2.5 Zusammenfassung

Die Annotierung der Daten des BoRIS-Experiments deckte auf, dass die Fehlerklassifikation in ihrer leicht veränderten Form auch für dieses System relevant ist, insofern als alle Klassen bis auf REFERENCE mindestens einmal annotiert wurden. Die Instanzen der Klassen sind jedoch teilweise sehr unterschiedlich. Ungefähr die Hälfte der Klassen war sehr selten.

Die Analyse der Daten wurde methodisch wie die Analyse des INSPIRE-Systems angegangen, so dass die Ergebnisse beider Untersuchungen vergleichbar sind. Die Faktorenanalyse ergab für die Beurteilung von BoRIS ähnliche Dimensionen wie bei INSPIRE, was insbesondere für die Faktoren Quality Of Information Exchange, COGNITIVE EFFORT und SPEED gilt. Die Berechnung der Korrelationen ergab prozentual etwa doppelt so viele für beide Koeffizienten signifikante Ergebnisse wie bei INSPIRE, die jedoch relativ niedrige Werte aufwiesen ($0{,}127 < |r| < 0{,}486$; $0{,}124 < |\rho| < 0{,}498$). Der größere Anteil an signifikanten Werten kann auf die größere Anzahl an Fällen zurückgeführt werden.

Die gefundenen Korrelationen ließen sich alles in allem besser erklären als beim INSPIRE-Experiment. So wurden nur wenige positive Korrelationen gefunden. Wie bei INSPIRE weisen auch für BoRIS die Konsequenzen insgesamt die höchsten Korrelationen mit den Benutzerurteilen auf, wobei im Speziellen STAGNATION und deren Sonderfall REPETITION am höchsten mit den Faktoren korrelieren.

5 Vorhersage von Benutzerurteilen auf Grundlage von Fehlern

5.1 Vorhersagemodelle

Nachdem in den vorangegangenen zwei Kapiteln gezeigt worden ist, dass zwischen den in einem Dialog auftretenden Fehlern und den Benutzerurteilen Zusammenhänge existieren, soll nun versucht werden, die Urteile mittels multivariater Verfahren aus den Fehlerhäufigkeiten vorherzusagen.

Compagnoni (2006) schlägt in seiner Diplomarbeit verschiedene Verfahren vor, von denen das Erstellen von Entscheidungsbäumen als favorisierte Methode hervorgeht, da hier zum einen eine relativ gute Vorhersagegenauigkeit gegeben ist und zum anderen anhand des Bildes von dem resultierenden Baum die Auswirkungen der verschiedenen Eingangsvariablen auf die Entscheidung ersichtlich sind. Im Zusammenhang der hier vorgesehenen Vorhersage der Benutzerurteile aus den Fehlerzahlen ist eine weitere günstige Eigenschaft der Entscheidungsbäume, dass sie keine linearen Beziehungen zwischen den Prädiktoren und der Zielvariablen voraussetzen. Obwohl die bisherigen Analysen des Zusammenhangs auch die Pearson-Korrelation berücksichtigten, die einen linearen Zusammenhang misst, deuten die vergleichsweise hohen Werte für Spearmans ρ an, dass ein stärkerer nichtlinearer Zusammenhang existiert[28]. Entscheidungsbäume sind also die beste Wahl für diese Daten und sollen deshalb als Vorhersagemodell verwendet werden.

5.1.1 Entscheidungsbäume

Bei der Erstellung eines Entscheidungsbaums wird der Datensatz derart in zwei Teile gespalten, dass die Summe der quadrierten Abweichungen vom Mittelwert einer der Vorhersagevariablen für beide Partitionen minimal wird. Jede Partition wird daraufhin nach dem selben Prinzip weiter geteilt, bis die verbleibenden Daten eine festgesetzte

[28] Üblicherweise sind diese etwas niedriger als *r*, hier jedoch häufig ungefähr gleich hoch.

minimale Größe unterschreiten. Die Teilungspunkte nennt man Knoten. Wird ein Knoten nicht weiter gespalten, bezeichnet man ihn als Blatt. An den Blättern lässt sich der vorhergesagte Zielwert für einen Zweig (also die vorangehenden Entscheidungen) ablesen. Die Werte der Zielvariablen können entweder kontinuierlich (Regressionsbäume) oder nominal (Klassifikationsbäume) sein.

Da mit dem Entscheidungsbaum letztlich Vorhersagen für unbekannte Fälle gemacht werden sollen, darf dieser nicht zu exakt auf die Trainingsdaten zugeschnitten sein. Deshalb ist es nicht sinnvoll, die Zweige so lange aufzuspalten, dass jedes Blatt nur einen einzelnen Fall repräsentiert. Alternativ dazu, dass man eine größere Zahl von Fällen als Minimum für ein Blatt spezifiziert, kann man den Baum auch zuerst vollständig berechnen, und daraufhin stutzen, d. h., eine festgelegte Anzahl von Knoten an den Enden entfernen.

5.1.2 Grundlegendes zum Verfahren

Entscheidungsbäume ermöglichen Vorhersagen für Daten, die unter ähnlichen Bedingungen gesammelt worden sind wie die Trainingsdaten für den Baum. Man kann also nicht ohne weiteres Vorhersagen für Interaktionen mit einem anderen System, einen anderen Kontext oder einer anderen Systemkonfiguration machen. Dies gilt insbesondere, wenn die für das Training zur Verfügung stehenden Daten relativ gering sind, wie im Fall von BoRIS und INSPIRE. Deshalb wird zum Testen der Vorhersagegenauigkeit ein Teil des Datenkorpus als Testkorpus verwendet. Da die Vorhersage nur Sinn macht, wenn die Testdaten unabhängig von den Trainingsdaten sind, müssen die Testfälle vom Training ausgeschlossen werden.

Um dennoch möglichst viel der Daten zu nutzen, um die Vorhersagegenauigkeit des Algorithmus zu errechnen, wurde die „Leave on out“-Methode benutzt. Dabei wird für jeden Benutzer als Testfall aus den verbleibenden Benutzern (als Trainingsfälle) ein Entscheidungsbaum errechnet, so dass es für jeden Benutzer ein aus den anderen Benutzern vorhergesagtes Urteil gibt. Das Urteil jeder Versuchsperson wird also auch aus einem anderen Baum berechnet. Um einen Wert für die Vorhersagegenauigkeit der

Methode zu erhalten, werden die vorhergesagten Werte jeweils mit den tatsächlichen Werten für jeden Probanden verglichen. Dafür stehen unterschiedliche Verfahren zur Verfügung.

Handelt es sich um einen Regressionsbaum, werden die Pearson-Korrelation, R^2 und „angepasstes“ R^2 für die vorhergesagten und die tatsächlichen Urteile berechnet. Für die Berechnung von R^2 wird die Quadratsumme der Vorhersagefehler für jeden der Fälle geteilt durch die Quadratsumme der Abweichungen der tatsächlichen Urteile von ihrem Mittelwert. Das Ergebnis wird von 1 subtrahiert.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \hat{e}_i^{\,2}}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

Beim angepassten R^2 wird zudem die Anzahl der Prädiktoren p berücksichtigt.

$$\overline{R}^2 = 1 - \frac{(n-1)}{(n-p)}(1 - R^2)$$

Der (angepassten) R^2-Wert lässt sich so interpretieren, dass er den prozentualen Anteil der durch die Prädiktoren erklärten Varianz der Zielvariablen darstellt.
Im Fall der Klassifikationsbäume wird die Übereinstimmung zwischen dem vorhergesagten und dem tatsächlichen Urteil durch den Anteil richtiger Vorhersagen ausgedrückt.

5.1.3 Wahl der Prädiktoren

In seiner Diplomarbeit schreibt Compagnoni (2006), dass die Prädiktoren zum einen mit den vorherzusagenden Urteilen möglichst hoch korrelieren sollten, zum anderen aber idealerweise untereinander möglichst geringe Korrelationen aufweisen, also unabhängig voneinander sein sollen. Bereits in (informellen) Vorversuchen wurde festgestellt, dass das

zweite Kriterium vernachlässigt werden kann, da es sich nicht in jedem Fall bewahrheitete.

Für das INSPIRE System wurde entschieden, Modelle für den Faktor COMMUNICATION WITH THE SYSTEM der ersten Systemmetapher zu berechnen, da hier die meisten und die höchsten Korrelationen gefunden wurden. Zudem enthält diese Dimension das Urteil zum Gesamteindruck des Benutzers, das als relevant für die Usability des Systems angesehen wird, welche wiederum von besonderem Interesse bei der Vorhersage ist. Es muss hier angemerkt werden, dass sich durch die Verwendung von Daten nur einer Systemmetapher die berücksichtigten Dialoge nach Abzug der unvollständigen Fälle auf 19 reduzieren. Obwohl keine diesbezügliche Bemerkung in der konsultierten Literatur gefunden wurde, ist doch anzunehmen, dass eine geringe Anzahl an Fällen ein weniger signifikantes Ergebnis zur Folge hat, d.h., dass mglw. zufällige Eigenschaften des Datensatzes das Ergebnis bestimmen, wodurch sich dessen Aussagekraft (auch für die Beschreibung desselben Datensatzes) relativiert.

Für BoRIS wurde als interessantester Faktor der erste, Quality of Information Exchange, der u. a. die Urteile über die Zufriedenheit des Benutzers und den Gesamteindruck enthält, ausgewählt. Im vorherigen Kapitel wurden sieben korrelierende Fehlerklassen für diesen Faktor berichtet, die in den folgenden Versuchen in unterschiedlichen Kombinationen als Prädiktoren genutzt werden sollen.

5.2 Ergebnisse

Die in diesem Kapitel berichteten Berechnungen basieren auf den MATLAB-Programmen, die Compagnoni im Rahmen seiner Diplomarbeit erstellt hat. Diese wurden zum Teil leicht modifiziert, um eine Reihe unterschiedlicher Kombinationen von Prädiktorvariablen mit verschieden stark beschnittenen Bäumen zu testen. Die Annahme, dass untereinander niedriger korrelierende Prädiktoren eine höhere Vorhersagekraft besitzen, bestätigte sich bei ersten Versuchen mit dem Algorithmus nicht, so dass diesbezüglich keine Einschränkungen für die Variablenkombinationen getroffen werden

sollen. Der Grad der Stutzung des Baumes, der zu einem optimalen Ergebnis führt, ist von der Anzahl und Art der Variablen abhängig und kann nicht durch Überlegungen gefolgert werden. Die Art der Stutztechnik folgt der Ausgangskonfiguration der MATLAB-Programme Compagnonis. Für die Regressionsbäume sollen die Bäume bis zu einer vorgegebenen Anzahl an Ebenen zurückgestutzt werden, während bei den Klassifikationsbäumen ein prozentualer Anteil der Astlänge weggenommen wird. In Fall der Klassifikationsbäume kann der Diplomarbeit entnommen werden, dass dies das empfohlene Verfahren sei, da es die besten Ergebnisse bringt.

5.2.1 Regressionsbäume

5.2.1.1 INSPIRE-System

Da für das INSPIRE-System die Korrelationen wesentlich höher waren, wenn die Daten nach Metaphern (Systemkonfigurationen) getrennt analysiert werden, sollen auch für die Vorhersagemodelle zuerst für eine einzelne Metapher berechnet werden. Da Metapher 1 am besten mit den Benutzerurteilen korrelierte, insbesondere der Faktor COMMUNICATION WITH THE SYSTEM viele signifikante Werte aufwies, soll dieser als Zielvariable für die Untersuchung dienen. Hier seien nochmals die signifikanten Beziehungen aufgelistet:

- INADEQUATE MAGNITUDE OF CONTROL (r=-0,59129; ρ=-0,51083)
- NOUN (r=-0,60512; ρ=-0,57005)
- SPACE MODELLING (r=-0,44123; ρ=-0,4879)
- REFERENCE (r=0,477702; ρ=0,493091)
- STAGNATION (r=-0,57424; ρ=-0,53345)
- REPETITION (r=-0,61457; ρ=-0,55185)
- HELP (r=-0,5134; ρ= -0,48421)

Zuerst soll eine Vorhersage mit allen sieben Fehler- und Konsequenzklassen versucht werden, wobei für das Stutzen verschiedene Anzahlen an zurückbleibenden Stufen getestet werden. Ein Test des Programms ergibt eine ungefähre Anzahl von 16 Stufen, so dass versuchsweise bis 15 verbleibende Stufen getestet wird, was funktioniert. Das beste Ergebnis ergibt sich für 2 verbleibende Stufen (r=0.6908; R^2=0.4365; angepasstes R^2=

0.0778). Der angepasste R^2-Wert ist sehr klein, was an der geringen Anzahl an Fällen relativ zu den Prädiktoren liegt. Bereits ab drei verbleibenden Stufen pendelt sich der Wert für *r* bei ca. 0,65 ein. R^2 fällt zunächst, steigt dann wieder an, erreicht den angegebenen Wert jedoch nicht mehr. Das angepasste R^2 fällt schnell in den negativen Bereich. Der Wert *r*=0,69 ist unter normalen Umständen als zufrieden stellend zu bewerten, jedoch ist das Ergebnis angesichts der sehr hohen Korrelation der REPETITION-Konsequenzklasse mit der Zielvariablen zu relativieren.

levelsleft = 1	correltest = -1 rsquaretest =-0.1142 adjrsquaretest =-0.8232
levelsleft = 2	correltest = 0.6908 rsquaretest = 0.4365 adjrsquaretest = 0.0778
levelsleft = 3	correltest = 0.6573 rsquaretest = 0.3464 adjrsquaretest =-0.0695
levelsleft = 4	correltest = 0.6528 rsquaretest = 0.3056 adjrsquaretest =-0.1363
levelsleft = 5	correltest = 0.6398 rsquaretest = 0.2446 adjrsquaretest =-0.2361
levelsleft = 6	correltest = 0.6361 rsquaretest = 0.2031 adjrsquaretest =-0.3040

Tabelle 5.1 Ergebnisse der Berechnung von Regressionsbäumen für alle sieben signifikant mit COMMUNICATION WITH THE SYSTEM korrelierenden Fehlerklassen. „Levelsleft" gibt die Anzahl der Entscheidungsstufen an.

Es soll nun versucht werden, durch Auslassung eines Prädiktors eine bessere Korrelation zwischen den vorhergesagten und den tatsächlichen Werten zu erhalten. Dazu soll zunächst ein Versuch mit sechs Fehlerklassen unternommen werden, wobei die Auslassung aller sieben Klassen nacheinander getestet werden soll (die detaillierten Ergebnisse finden sich im Appendix (0)). Das beste Ergebnis zeigt sich überraschenderweise unter Auslassung der Konsequenz REPETITION, die den höchsten Korrelationswert mit dem Faktor aufweist. Sowohl für *r* als auch für R^2 ergeben sich sehr

hohe Werte von 0,76 bzw. 0,575. Wiederum wird mit zwei Stufen nur ein sehr kleiner Teil des Regressionsbaumes für die optimale Berechnung verwendet. Ab vier Stufen werden die Werte deutlich niedriger.

Das Ergebnis ermutigt zu einem Versuch mit fünf Prädiktoren. Wieder sollen alle möglichen Kombinationen aus zwei Prädiktoren einmal weggelassen und verschieden starke Stutzungen probiert werden. Hier zeigt sich ein wesentlich weniger systematisches Bild als bei den vorherigen zwei Berechnungen, da die Werte für die verschiedenen Prädiktorkombinationen bei gleicher Anzahl Entscheidungsstufen sehr unterschiedlich sind, andererseits aber keine generelle Tendenz zu erkennen ist, wenn sich die Stufenzahl ändert. Dies könnte daran liegen, dass die Ergebnisse aufgrund der kleinen Anzahl an Fällen relativ zufällig sind. Immerhin zeigen sich aber tatsächlich noch größere Werte (r=0,8; R^2=0,59; angepasstes R^2=0,43) für die Berechnung ohne NOUN- und SPACE-MODELLING-Fehler bei zehn verbleibenden Entscheidungsstufen bzw. (r=0,8; R^2=0,56; angepasstes R^2=0,39) bei sechs verbleibenden Stufen (R^2 und angepasstes R^2 sind in diesem Fall niedriger als bei den besten Werten mit sechs Prädiktoren). Dass die zwei besten Vorhersagen mit der selben Prädiktorkombination erzielt wurden, spricht in jedem Fall gegen ein zufälliges Ergebnis. Da die Werte bisher kontinuierlich höher wurden, muss an dieser Stelle auch ein Versuch mit vier Vorhersagegrößen unternommen werden!

In der Tat ergibt sich eine noch bessere Korrelation zwischen den vorhergesagten und den tatsächlichen Urteilen, (r=0,85; R^2=0,68; angepasstes R^2=0,59), wenn neben NOUN- und SPACE-MODELLING-Fehlern auch die Konsequenz REPETITION aus der Liste der Prädiktoren gestrichen wird und der Baum auf fünf Stufen zurückgestutzt wird. Der Wert sinkt für höhere Anzahlen an Entscheidungsstufen nur schwach ab, was für eine relativ solide Berechnung spricht. Auch sind alle drei Prädiktoren, die für die optimale Vorhersagekraft weggelassen werden müssen, bereits in den vorigen Berechnungen als ungeeignete Vorhersagevariablen aufgefallen.

Für die Vorhersage aus nur noch drei Variablen ergibt sich schließlich keine Steigerung der Übereinstimmungsmaße mehr. Zwar erreicht die Vorhersage hier noch eine relativ hohe Genauigkeit, die in einem Pearson-Koeffizienten von r=0,83 (R^2=0,64; angepasstes

R^2=0,56) resultiert, wenn zusätzlich zu NOUN- und SPACE-MODELLING-Fehlern und der REPETITION-Konsequenz noch die HELP-Konsequenz ausgesondert wird, jedoch übersteigt dieser Wert den vorherigen aus der Prädiktion mit vier Variablen nicht. Nicht zuletzt, da die Tendenz rückgängig ist, sind bessere Vorhersagen mit noch weniger Variablen nicht zu erwarten, weshalb die Analyse hier abgebrochen werden soll. Mit den Werten r=0,85 bzw. R^2=0,68 bzw. angepasstes R^2=0,59 sind sehr zufrieden stellende Ergebnisse gefunden worden! (Compagnoni (2006) erreichte für das INSPIRE-System mit Regressionsbäumen eine Übereinstimmung von maximal r=0,60 bei der Vorhersage von Einzelurteilen aus geloggten Systemparametern!)

5.2.1.2 BoRIS-System

Für BoRIS wurde zuerst eine Vorhersage für den Faktor Quality of Information Exchange, ebenfalls mit allen sieben korrelierenden Fehlerklassen versucht. Diese seien hier nochmals wiederholt:

- EXTERNAL RESTRICTIONS (r= -0,276; ρ= -0,258)
- ATTRIBUTE TYPE (r= -0,222; ρ= -0,217)
- TIME MODELLING (r= -0,244; ρ= -0,230)
- STAGNATION (r= -0,219; ρ= -0,218)
- REPETITION (r= -0,186; ρ= -0,176)
- REGRESSION (r= -0,486; ρ= -0,498)
- RESTART (r= -0,478; ρ= - 0,435)

Die Anzahl der verbleibenden Stufen nach dem Stutzen wurde von 1 bis 50 variiert, wobei sich das beste Ergebnis mit drei Entscheidungsstufen ergab. Ein Bild des letzten berechneten Baumes (für VP 40 als Testfall) zeigt, dass die beiden am höchsten korrelierenden Klassen REGRESSION und RESTART für die Berechnung herangezogen werden. In Anbetracht dessen, dass beide Klassen jeweils alleine fast die gleiche Vorhersagekraft besitzen wie der Baum, ist das Ergebnis eher enttäuschend.

levelsleft = 1	correltrain = 0.7937 rsquaretrain = 0.6299 adjrsquaretrain = 0.6163 correltest =-0.5987 rsquaretest =-0.0195 adjrsquaretest =-0.0615
levelsleft = 2	correltest = 0.2652 rsquaretest = 0.0347 adjrsquaretest =-0.0050
levelsleft = 3	correltest = **0.5266** rsquaretest = **0.2756** adjrsquaretest = **0.2457**
levelsleft = 4	correltest = 0.4994 rsquaretest = 0.2419 adjrsquaretest = 0.2106
levelsleft = 5	correltest = 0.4831 rsquaretest = 0.2134 adjrsquaretest = 0.1810
levelsleft = 6	correltest = 0.4542 rsquaretest = 0.1636 adjrsquaretest = 0.1291

Tabelle 5.2 Ergebnisse der Berechnung von Regressionsbäumen für alle sieben signifikant mit Quality of Information Exchange korrelierenden Fehlerklassen.

Da Compagnoni (2006) darauf hinweist, dass teilweise bessere Ergebnisse gefunden werden, wenn weniger Prädiktoren genutzt werden, soll als nächstes versucht werden, eine bessere Vorhersage aus sechs der Fehler- bzw. Konsequenzklassen zu berechnen. Dafür sollen wiederum alle möglichen Kombinationen der Klassen getestet werden. Für das Stutzen sollte eine Anzahl von acht verbleibenden Stufen aufwärts nicht mehr interessant sein, wie das vorherige Ergebnis schon zeigte.

Die größten derart gefundenen Werte entsprechen jenen aus der optimalen Berechnung mit sieben Variablen. Lediglich das angepasste R^2 ist etwas größer, da hier die Anzahl der Prädiktoren im Nenner einfließt. Die Tatsache, dass die Ergebnisse bei der entsprechenden Anzahl von drei Stufen immer gleich sind, sofern nicht REGRESSION oder RESTART

ausgeschlossen werden, deutet darauf hin, dass auch in diesem Fall die Berechnung ausschließlich auf diesen beiden Klassen fußt, sofern sie zur Verfügung stehen.

Trotz dieses Ergebnisses soll noch eine Vorhersage aus fünf Variablen versucht werden. Wieder werden alle möglichen Kombinationen mit fünf Fehlerklassen für verschieden viele verbleibende Entscheidungsstufen getestet. Wie sich überraschenderweise zeigt, lässt sich derart das Ergebnis leicht verbessern. Während bei drei verbleibenden Entscheidungsstufen noch das gleiche Ergebnis wie zuvor erzielt werden kann, Lässt sich mit vier Stufen ein maximaler Werte für die Messgrößen feststellen, wenn man die Klassen STAGNATION und REPETITION, die von den sieben Klassen am schwächsten mit dem Faktor Quality of Information Exchange korrelieren, nicht für die Berechnung berücksichtigt. Es ergeben sich die Werte r=0,5547, R^2=0,3064 und angepasstes R^2=0,2863.

Das gleiche Ergebnis erhält man auch, wenn man die Vorhersage aus vier Prädiktoren versucht. Dabei spielt es keine Rolle, ob man EXTERNAL-RESTRICTIONS-, TIME-MODELLING- oder ATTRIBUTE-TYPE-Fehler als dritte Fehlerklasse aussondert. In allen Fällen ist der Wert für das angepasste R^2 etwas höher (0.2904), da ein Prädiktor weniger benutzt wurde als im vorherigen Fall. Im Gegensatz zu der Vorhersage aus fünf Prädiktoren ergibt die Vorhersage aus vier Variablen außerdem auch gute Ergebnisse für Bäume, die bis auf fünf Entscheidungsstufen gestutzt sind. Die optimale Prädiktorenkonstellation ist dabei die gleiche wie für vier Stufen.

Entfernt man einen weiteren Prädiktor, fällt das beste Ergebnis für Pearsons Korrelationskoeffizienten unter 0,5. Demnach kann erwartet werden, dass für weniger Prädiktoren die Vorhersagekraft des Modells für die BoRIS-Daten nicht gesteigert wird. Die gefundenen Ergebnisse sind, wenn auch nicht so gut wie für die INSPIRE-Daten, dennoch höher als die von Compagnoni (2006) berechnete beste Vorhersage aus Systemparametern für ein Einzelurteil von BoRIS (r=0,46).

5.2.2 Klassifikationsbäume

Um Klassifikationsbäume berechnen zu können, musste zuerst die kontinuierliche Zielvariable in eine kategoriale umgewandelt werden. Um eine eher geringe Ratewahrscheinlichkeit zu haben, wurden die Urteile in fünf Stufen untergliedert, und zwar so, dass alle Kategorien etwa gleich häufig vertreten sind. Die Grenzen diesen richten sich also nach der Häufigkeit der Bewertungen in einem bestimmten Bereich der Skala. Die Ratewahrscheinlichkeit ergibt sich bei fünf Stufen zu 1/5 bzw. 20 Prozent. Die Ergebnisse sollten idealerweise deutlich darüber liegen. Das Stutzen der Klassifikationsbäume soll hier prozentual vorgenommen werden, wobei es angemessen erscheint, in Zehn-Prozent-Schritten Vorhersagen zu errechnen.

5.2.2.1 INSPIRE-System

Mit den Regressionsbäumen wurde für INSPIRE das beste Ergebnis einer Vorhersage mit vier Prädiktoren erzielt, wobei NOUN- und SPACE-MODELLING-Fehler sowie die Konsequenz REPETITION nicht für die Prädiktion genutzt wurden. Da die Klassifikationsbäume nach dem selben Prinzip berechnet werden, der Unterschied nur in der Skalierung der Zielvariablen besteht, die hier nominal und nicht rational ist, soll der erste Versuch für dieses Vorhersagemodell mit eben dieser Konfiguration versucht werden. Es hat sich zudem für die Regresionsbäume gezeigt, dass sowohl bei Variation des Stutzungsgrades als auch der Prädiktorenanzahl die Ergebnisse für eine bestimmte Konfiguration am besten sind und von dort zu beiden Seiten abfallen. Deshalb soll diesmal, ausgehend von der geschätzten besten Konfiguration, durch Hinzunahme bzw. Aussonderung einer Variablen zunächst die „Umgebung“ der Konfiguration untersucht werden. Steigen die Messwerte in eine Richtung (also z. B. bei Hinzunahme einer Variablen), wird das Verfahren entsprechend weitergeführt. Fallen sie, so soll das Verfahren abgebrochen werden.

Anstatt Werte nur für diese spezielle Konfiguration zu berechnen, wurden gleich alle Varianten mit vier Prädiktoren für verschiedene große Maße an Stutzung der Bäume berechnet. Der größte Wert für die Übereinstimmung der vorhergesagten mit den tatsächlichen (kategorisierten) Urteilen, der unter den Ergebnissen zu finden ist, ist 0,5263.

Dieser entfällt jedoch nicht auf die genannte Kombination aus Vorhersagevariablen. Stattessen müssen für die entsprechende Vorhersage INADEQUATE MAGNITUDE OF CONTROL, REFERENCE und STAGNATION von der Vorhersage ausgeschlossen werden. Der Anteil an Verschnitt des Baumes kann von null bis 20 Prozent variieren, ohne dass das Ergebnis sich ändert. Dass der vollständig ausgearbeitete Baum die höchste Vorhersagegenauigkeit erbringt, macht etwas stutzig, da er in diesem Zustand eigentlich „overfitted" sein müsste. Da jedoch die Blätter des Baumes nicht übermäßig bestimmt sind (es gibt nur fünf Stufen), soll ein solches Ergebnis nicht ausgeschlossen werden. Die gefundene Vorhersagegenauigkeit stellt prinzipiell ein recht gutes Ergebnis dar, jedoch wurde in Anbetracht der sehr guten Ergebnisse mit den Regressionsbäumen eher ein höherer Wert erwartet. Die beste Vorhersage für die Variablenkombination, die mit den Regressionsbäumen zum optimalen Ergebnis führte, beträgt hier 0,26 (also 26 Prozent korrekte Vorhersagen), was die Ratewahrscheinlichkeit nur knapp überschreitet.

Lässt man eine weitere Variable zur Vorhersage weg, steigt die Vorhersagegenauigkeit leicht auf 0,58 an. Diesen Wert erhält man, wenn man zusätzlich zu INADEQUATE MAGNITUDE OF CONTROL, REFERENCE und STAGNATION auch REPETITION streicht und den Baum um 10 bis 20 Prozent zurückschneidet. Bei Auslassung einer weiteren Vorhersagevariablen fallen die Werte deutlich ab. Der beste derart gefundene Wert beträgt 0,37.

Obwohl schon jetzt vermutet werden kann, dass sich auch bei Hinzufügen weiterer Variablen die Ergebnisse eher verschlechtern werden, soll – sicherheitshalber – noch ein Versuch mit fünf Prädiktoren unternommen werden. Hier kann die maximale Vorhersagegenauigkeit unter Auslassung von SPACE-MODELLING- und REFERENCE-Fehler erreicht werden; sie beträgt 0,42. Auch dieser Wert liegt deutlich unter dem somit besten Ergebnis von 58 Prozent korrekter Vorhersagen, die mit den Prädiktoren NOUN- und SPACE- MODELLING-Fehler sowie HELP-Konsequenz erreicht werden kann. Die Übereinstimmung der Vorhersage mit der tatsächlichen Urteilsklasse von 58 Prozent kann als vergleichsweise gutes Ergebnis gewertet werden. Compagnoni (2006) kam mit der Vorhersage eines Einzelurteils aus den geloggten Systemparametern für das INSPIRE-

System je nach Systemmetapher auf 50 -60 Prozent, jedoch wurde hier das Urteil nur in drei Kategorien geteilt, so dass die Ratewahrscheinlichkeit bereits bei 33 Prozent lag.

5.2.2.2 BoRIS-System

Für BoRIS hatte sich mit den Regressionsbäumen die beste Vorhersage aus vier oder fünf Variablen ergeben. Obwohl sich soeben zeigte, dass sich die optimale Konfiguration für die Vorhersage durchaus unterscheidet, je nachdem, ob Regressions- oder Klassifikationsbäume berechnet werden, soll auch bei der Kalkulation von Klassifikationsbäumen für die BoRIS-Daten mit vier Prädiktoren angefangen werden, da die bisherigen Erfahrungen zeigen, dass dies eine realistische Schätzung ist.

Es ergibt sich ein maximales Ergebnis von 0,3989, das zum einen unter Auslassung von EXTERNAL RESTRICTIONS, ATTRIBUTE TYPE und RESTART zustande kommt. Wieder ist der Baum mit dem besten Ergebnis nur wenig (10%) zurückgeschnitten. Weiterhin kommt man zum selben Ergebnis, wenn neben EXTERNAL RESTRICTIONS und ATTRIBUTE TYPE entweder STAGNATION oder REPETITION nicht als Prädiktoren verwendet werden und der resultierende Baum 20 Prozent zurückgeschnitten wird.

Da zuvor die weitere Reduktion der Prädiktoren zum Ziel führte, soll dieser Weg auch diesmal beschritten werden. Tatsächlich führt er zu einem etwas höheren Wert für das Übereinstimmungsmaß: unter Verwendung der drei Konsequenzen REPETITION, REGRESSION und RESTART lassen sich 40,45 Prozent der Urteile korrekt vorhersagen, wenn der Entscheidungsbaum aus den Variablen um 30 Prozent gestutzt wird. Einige andere Kombinationen erreichen ungefähr 39 Prozent. In einem nächsten Schritt soll untersucht werden, ob sich der Trend bei zwei Prädiktoren fortsetzt.

Wie schon beim INSPIRE-System sinkt auch in diesem Fall die Prädiktionsgenauigkeit wieder ab, wenn ein weiterer Prädiktor gestrichen wird. Die beste Vorhersage ergibt sich mit den Klassen TIME MODELLING und REGRESSION zu 38,76 Prozent richtiger Vorhersagewerte (30/40 Prozent weggestutzt).

Auch für diesen Datensatz soll vorsichtshalber die These, dass die Funktion der Vorhersagegenauigkeiten über der Prädiktoranzahl eingipfelig ist, geprüft werden, indem auch für fünf Vorhersagevariablen eine Prädiktion versucht wird. Diesmal zahlt sich die Vorsicht aus, da der bisher größte Wert gefunden wird: 42,13 Prozent. Dieser kommt zustande, wenn die Klassen EXTERNAL RESTRICTIONS und REPETITION nicht als Prädiktoren verwendet werden und 30 Prozent der Entscheidungsknoten des Klassifikationsbaumes entfernt werden. Lässt man statt REPETITION STAGNATION für die Erstellung des Baumes beiseite, erreicht dieser immerhin noch eine Genauigkeit von 40 Prozent.

Da sich nun doch eine Tendenz zu besseren Vorhersagen bei mehr Prädiktoren andeutet, soll auch für sechs Eingangsgrößen errechnet werden, wie gut sich so idealerweise die Urteile vorhersagen lassen. Der beste Wert ergibt sich ohne ATTRIBUTE TYPE als Vorhersagegröße bei 40 Prozent Stutzung es Baumes zu 0,3876. Dies ist deutlich niedriger als der Wert für fünf Prädiktoren, weshalb eine Vorhersage aus allen sieben Fehlerklassen nicht mehr versucht werden soll.

Das Ergebnis von 42 Prozent korrekt vorhergesagter Urteile ist, wie bei den Regressionsbäumen, nicht so gut wie das Ergebnis für die Vorhersage des INSPIRE-Faktors. Der beste von Compagnoni (2006) errechnete Klassifikationsbaum für Einzelurteile des BoRIS-Systems konnte knapp 61 Prozent der Urteile korrekt prädizieren, jedoch ist der Wert dem hier gefundenen nicht direkt vergleichbar, da Compagnoni nur drei Klassen für die Urteile bildete. Vergleicht man beide Werte mit der entsprechenden Ratewahrscheinlichkeit, so lässt sich konstatieren, dass beide doppelt so gut vorhersagen wie eine zufällige Bestimmung der Klassen.

6 Conclusio

Diese Magisterarbeit widmete sich der Vorhersage von Benutzerurteilen aus Fehlern bei der Interaktion mit Sprachdialogdiensten. Zur Umsetzung dieses Ziels wurde zuerst anhand einer Analyse von in Experimenten geführten Dialogen mit dem INSPIRE-Smart-Home-System eine Klassifikation der Fehler und deren Konsequenzen ausgearbeitet. Die Transkriptionen aller 72 Dialoge aus dem Free-Wizard-of-Oz-Experiment wurden sodann mit der Fehlerklassifikation annotiert. Anschließend wurden die Fehlerhäufigkeiten zu den Benutzerurteilen in Beziehung gesetzt, wofür als Zusammenhangsmaße der Pearsonsche sowie der Spearmansche Korrelationskoeffizient verwendet wurden. Die Benutzerurteile wurden dafür zu Faktoren zusammengefasst, da sonst durch die hohe Anzahl an errechneten Werten viel Raum für zufällige Messergebnisse geblieben wäre. Die Berechnung wurde sowohl für den kompletten Datensatz als auch für die nach Systemkonfiguration getrennten Dialoge ausgeführt, wobei in ersterem Fall eher niedrige, im zweiten Fall jedoch recht hohe Korrelationswerte gefunden wurden.

Die Klassifikation der Fehler und ihrer Konsequenzen wurde anschließend nochmals daraufhin untersucht, ob sich durch Veränderungen höhere Korrelationen ergeben könnten, woraus einige leichte Veränderungen des Klassifikationsschemas hervorgingen. Mit dem resultierenden Schema wurden 200 Dialoge aus einem Experiment mit dem BoRIS-Restaurantinformationssystem annotiert. Dabei wurden fast alle Fehlerklassen wieder gefunden, jedoch traten einige der Fehler nur selten auf. Insgesamt wurden in diesen Dialogen weniger Fehler gefunden als in denen mit INSPIRE.

Für die Korrelationsberechnungen wurden wiederum Faktoren der Benutzerurteile verwendet, die teilweise denen für das INSPIRE-System ähnlich waren. Es fanden sich prozentual doppelt so viele signifikante Korrelationen mit den Bewertungsdimensionen wie beim INSPIRE-System, jedoch wiesen diese im Vergleich zu der metapherweisen Analyse der INSPIRE-Daten relativ niedrige Werte auf.

Um einen noch größeren Teil der Varianz der Benutzerurteile erklären zu können, wurden schließlich Regressions- und Klassifikationsbäume für die Vorhersage der

Bewertungsdimensionen beider Systeme erstellt, wofür von Compagnoni (2006) erstellte MATLAB-Programme adaptiert wurden.

Die Berechnung von Regressionsbäumen für das INSPIRE-System ergab die besten Vorhersagen für den Urteilsfaktor COMMUNICATION WITH THE SYSTEM, wenn die Fehlerklassen NOUN und SPACE-MODELLING, sowie die Konsequenz REPETITION als Prädiktoren dienten. Die Pearson-Korrelation der vorhergesagten Urteile mit den tatsächlichen Urteilen erreicht so einen maximalen Wert von 0,85. Die Klassifikationsbäume sagten 58 Prozent der Faktorwerte richtig vorher, wenn die Fehlerklassen NOUN und SPACE-MODELLING, sowie die Konsequenz HELP als Prädiktoren verwendet wurden. Die Urteile waren in diesem Fall in fünf etwa gleich große Klassen unterteilt, so dass die Ratewahrscheinlichkeit lediglich 20 Prozent betrug. Die Ergebnisse beider Vorhersagen wurden als sehr zufrieden stellend beurteilt.

Für BoRIS waren die Ergebnisse der Vorhersagen etwas weniger gut, jedoch konnte letztendlich auch hier Zufriedenheit konstatiert werden. Für die Regressionsbäume ergab sich eine Pearson Korrelation mit r=0,55, wenn aus den Klassen EXTERNAL RESTRICTIONS, ATTRIBUTE TYPE, TIME-MODELLING, REGRESSION und RESTART vorhergesagt wurde, wobei eine der drei Fehlerklassen auch weggelassen werden kann. Bei den Klassifikationsbäumen führte die Berechnung zu einer Übereinstimmung von 42 Prozent zwischen den vorhergesagten und den tatsächlichen Urteilen, wenn alle Klassen außer EXTERNAL RESTRICTIONS und REPETITION als Prädiktoren verwendet wurden.

Konsequenzen für die Vorhersage von Benutzerverhalten

Die hier beschriebene Arbeit soll in einem Projekt der Deutsche Telekom Laboratories weitergeführt werden, wobei vorgesehen ist, Fehler in einer Simulation von Interaktionen mit einem im Entwicklungsprozess befindlichen System zu generieren, um deren Auswirkungen auf die Systemhandhabung und letztendlich die Usability-Beurteilung vorhersagen zu können. Bei der Simulation des Benutzers sollen auch die in dieser Arbeit beschriebenen Benutzerfehler automatisch generiert werden, wobei das Mentale Modell des Benutzers herangezogen werden soll, um das Auftreten eines Fehlers vorherzusagen.

Die Bemerkungen in Kapitel 3.1 zu Mentalen Modellen machen einige grundlegende Schwierigkeiten dieses Ansatzes deutlich. So sind Mentale Modelle für verschiedene Benutzer unterschiedlich, d. h., sie müssen für jeden Benutzer bzw. jede Gruppe von Benutzern neu gefunden und (im Vorhersagealgorithmus) definiert werden. Zudem sind die Modelle dynamisch, d. h., das Verhalten eines Benutzers kann nicht mit einem einzigen Mentalen Modell erklärt werden. Die Entwicklung eines Mentalen Models ist zudem von den Informationen abhängig, die der Benutzer vor oder während der Interaktion über das System bekommt. Die in dieser Arbeit behandelten Systeme erheben den Anspruch, von einem Benutzer ohne Vorwissen bedienbar zu sein, was in diesem Zusammenhang bedeutet, dass das Mentale Modell außer durch die Übertragung von Wissen aus anderen Situationen vor allem durch die Systemreaktion bzw. die Systemausgabe gebildet wird. Sollen realistisch Fehler vorhersagende Modelle gebildet werden, wäre es demnach sinnvoll, den Einfluss der Systemreaktion auf das Modell zu untersuchen. Dafür bieten die in Kapitel 3.1 beschriebenen Methoden zur Identifikation der Modelle einen Ansatzpunkt.

Die von Norman (1983) und anderen beschriebene Eigenschaft Mentaler Modelle, in der Regel inkonsistent, unvollständig usw. zu sein, wird die Identifikation erschweren und bei der algorithmischen Beschreibung der Benutzer eine gewisse Unschärfe zurücklassen. Ohne hier tiefer auf Möglichkeiten und Zweck von Benutzersimulationen einzugehen, soll an dieser Stelle zumindest bemerkt sein, dass eine Annäherung an das tatsächliche Verhalten von Benutzern für bestimmte Zwecke, z. B. die Tauglichkeit der Fehlerhandhabungsstrategie des Systems, ausreichend sein kann. Zudem wird das Auftreten von Fehlern ohnehin nicht allein durch Diskrepanzen zwischen dem Mentalen Modell und dem tatsächlichen System erklärt, sondern ist von u7nterschiedlichen Faktoren wie z. B. der Konzentration des Benutzers abhängig[29].

Die Problematik der Generierung plausibler Benutzerfehler wirft die Frage auf, ob eine Genotypische Klassifikation nicht prinzipiell sinnvoller wäre als eine phänotypische. Als Begründung für die Wahl letzterer Methode wurde oben (Kapitel 4.2) angegeben, dass

[29] Vgl. hierzu die Unterscheidung zwischen „Slips“ und „Mistakes“, Kapitel 3.2

Informationen über die Entstehung der Fehler nicht in genügendem Maße vorhanden gewesen seien. In neuen Experimenten könnte jedoch versucht werden, mit Hilfe der beschriebenen Methoden zur Identifizierung Mentaler Modelle diese zu erfassen und einer entsprechenden Klassifikation zugrunde zu legen. Ein bereits existierendes genotypisches Fehlerklassifikationsschema ist in der ISO-Norm 9241-110 gegeben, die zwischen den Fehlern (bzw. Problemen) Suitability for the task, Controllability, Self-descriptiveness, Error Tolerance, Suitability for Individualisation, Suitability for learning und Conformity with user expectation unterscheidet. Diese Klassen können für die meisten interaktiven Systeme sinnvoll verwendet werden, beschreiben jedoch offensichtlich das System, während für die Modellierung eines Benutzers das Benutzerverhalten kategorisiert werden sollte, denn dieses soll ja im Modell erzeugt werden. Die Klassen aus der ISO-Definition können mglw. aber als Prädiktoren für die Wahrscheinlichkeiten des Auftretens der phänotypischen Fehler, wie sie hier beschrieben wurden, dienen. Hierzu muss angemerkt werden, dass das Auftreten eines Systemproblems nach ISO nur schwerlich automatisch detektiert werden kann, so dass auch hier als Grundlage der Arbeit mit den Fehler-Phänotypen einige manuelle Arbeit des Testers nötig wäre. Der Nutzen dieses Verfahrens bestünde jedoch darin, den Einfluss eines Verstoßes gegen die ISO-Richtlinien auf die Benutzbarkeit des Systems untersuchen zu können. Somit können z. B. Design-Präferenzen gegen eine optimale Handhabung abgewogen werden.

Zukünftige Arbeiten

Im Kontext dieser Perspektive ergeben sich aus der vorgelegten Arbeit kurzfristige und langfristige Vorhaben.

Zunächst sollen weitere Vorhersagemodelle bezüglich ihrer Prädiktionsgenauigkeit getestet werden. Insbesondere sind hier im Moment die auch von Compagnoni (2006) verwendeten Neuronalen Netze angedacht, da diese mglw. komplexere Beziehungen auch der Prädiktoren untereinander verarbeiten können. Zusätzlich zu den Fehlerklassen sollen auch ausführliche Versuche mit den Systemparametern, die bei den Versuchen geloggt wurden, durchgeführt werden, wobei auch die Kombination dieser mit den Fehlerklassen interessant sein wird.

Ein weiteres Ziel besteht darin, die Fehlerklassifikation für weitere Systeme, insbesondere solche, die eine deutlich andere Funktionsweise oder gar Interaktionsform aufweisen (z. B. Webseiten), angepasst werden. In diesem Zusammenhang wird eine Herausorderung auch darin bestehen, die vorhandenen und neuen Klassen so zusammenzufassen, dass bei der Annotation genügend Fehler auf jede Gruppe abfallen, um statistische Berechnungen sinnvoll durchführen zu können. Dabei ist selbstredend ein entscheidendes Kriterium, dass diese neuen Klassen (oder die Gruppen von Klassen) gleich gute oder sogar höhere Korrelationen mit den Benutzerurteilen aufweisen. Ein erster Schritt bei der sinnvollen Gruppierung der Klassen wurde bereits im Rahmen dieser Arbeit getan, indem die bei der Annotation der INSPIRE-Daten definierten Klassen sinnvoll der Ziel-, Aufgaben, Modellierungs- oder Kommandoebene zugeordnet wurden (vgl. Kapitel 3.4.2). Erste Versuche mit diesen Klassengruppen aus den BoRIS-Daten, die hier nicht dokumentiert sind, zeigen, dass diese gut mit den Werten der Urteilsdimensionen korrelieren.

Langfristig erscheint es, wiederum im Kontext des genannten Projektes der Deutsche Telekom Laboratories, sinnvoll, die Auftretenswahrscheinlichkeit der Fehler möglichst weitgehend automatisch vorherzusagen. Dazu müssen die Zusammenhänge zwischen den Systemeigenschaften (ausgedrückt z. B. in der ISO-Problemklassifikation oder den Interaktionsparametern der Log-Dateien) und den Fehlern zu erforschen. Dabei wären vor allem Beziehungen interessant, die für unterschiedliche Systeme Gültigkeit besitzen. Denkbar wären Zusammenhänge der Art „Wenn der System-Prompt mehr als 30 Wörter beinhaltet, tritt mit 40% Wahrscheinlichkeit ein REGRESSIVE-DECISION-Fehler auf“. Aufgrund der Heterogenität der Modelle wird eine Herausforderung dieser Aufgabe darin bestehen, sinnvolle Gruppen aus den Benutzern zu bilden, für die das Zusammenspiel aus Voraussetzungen und Entwicklung des Mentalen Modells hinreichend ähnlich ist.

Für die Suche nach solchen Zusammenhängen könnten auch Methoden aus der Forschung an Mentalen Modellen hilfreich sein, da auch diese versucht, Verhalten von Menschen bei der Interaktion mit Maschinen vorherzusagen und (phänotypisch erfasste) Fehler des Benutzers mit Begründungen (in diesem Falle jedoch mittels kognitiver Prozesse anstatt von Eigenschaften der Maschine) in Beziehung zu setzen. So könnte zum Beispiel in

Anlehnung an die Vorhersage von Verhalten aus einem antrainierten (oder am bisherigen Verhalten abgeschätzten) kognitiven Status die Fehlerhäufigkeiten bei verschiedenen Systemkonfigurationen (z. B. mit unterschiedlich langen Prompts) getestet werden.

Die in dieser Arbeit berichteten Ergebnisse der bisherigen Analysen stimmen den Autor zuversichtlich, einige der genannten Ziele in den nächsten Jahren zu verwirklichen.

7 Appendix

7.1 Szenariobeschreibungen INSPIRE

Szenario A

Situation:

Nach einer turbulenten Woche mit vielen Terminen möchten Sie und Ihr Partner sich einen ruhigen Abend zu Hause gönnen. Ihnen schwebt zunächst ein schönes Abendessen vor und danach vielleicht ein netter Film im Fernsehen. Während Ihr Partner sich schon um das Essen kümmert, **(1) schauen Sie in die elektronische Programminformation**, um sich einen Überblick über das Abendprogramm zu verschaffen. Sie sitzen dabei entspannt auf der Couch und **(2) wählen aus den Angeboten, die Ihnen das System auf dem Bildschirm macht**. Da sich Ihr Partner gerade in der Küche befindet, wird er durch Ihr Sprechen nicht gestört. Sobald Sie einen interessanten Spielfilm gefunden haben, **(3) bitten Sie das INSPIRE-System, Sie rechtzeitig an den Beginn des Films zu erinnern**.

Es war ein sonniger und heißer Tag. Da es immer noch recht warm im Wohnzimmer ist, **(4) benutzen Sie den Ventilator, um etwas Frischluft zu bekommen** und **(5) das Rollo, um die Sonne abzublenden**. Ihr Partner ist immer noch mit dem Essen beschäftigt, und deshalb **(6) fragen Sie noch Ihren Anrufbeantworter ab**. Sie haben eine Nachricht von einer Person, die sich wie ein Verkäufer anhört; deshalb entschließen Sie sich, **(7) diese Nachricht nicht zu speichern**. Da es nun kühl genug ist, **(8) benötigen Sie den Ventilator nicht mehr**.

Es wird langsam dunkel. Sie **(9) sorgen mit zwei Lampen für etwas Licht** und **(10) reduzieren deren Helligkeit**. Später am Abend entscheiden Sie sich, den Film doch nicht anzuschauen, sondern stattdessen ein gutes Buch zu lesen. Sie **(11) bitten das INSPIRE-System, den Film für später aufzuzeichnen**.

Szenario B

Situation:

Es ist 17 Uhr 30. Sie und Ihr Partner kommen von einem schönen, aber auch anstrengenden Tagesausflug nach Hause. Es war ein heißer und sonniger Tag und das Wohnzimmer hat sich stark

aufgeheizt; deshalb entschließen Sie sich, **(1) mit dem Ventilator für etwas frische Luft zu sorgen** und **(2) die Sonne mit dem Rollo etwas abzublenden**.

Da Sie den ganzen Tag unterwegs waren nehmen Sie an, dass Ihre Tochter eine Nachricht auf dem Anrufbeantworter hinterlassen hat, um einen Termin für das Treffen übermorgen abzumachen. Sie sitzen entspannt auf Ihrer Couch und **(3) bitten das INSPIRE-System, Ihren Anrufbeantworter abzufragen**. In der Tat finden Sie eine Nachricht Ihrer Tochter, die einen Zeitpunkt vorschlägt. Eine zweite Nachricht stammt von einem Verkäufer, der irgendein Produkt anpreist, an dem Sie aber nicht interessiert sind. Daher **(4) sichern Sie die Nachricht Ihrer Tochter Maria**, **(5) nicht jedoch die zweite Nachricht**. Es ist nun wieder kühl genug, sodass **(6) der Ventilator überflüssig geworden ist**.

Während Ihr Partner in der Küche das Essen zubereitet, **(7) möchten Sie die „heute nachrichten" sehen**. Sie möchten auch **(8) erfahren, welche Sendungen heute Abend im Fernsehen laufen**. Es wäre doch schön, bei einem guten Glas Wein einen entspannenden Film zu sehen. Deshalb konsultieren Sie die elektronische Programminformation. Sie **(9) schauen in die Liste, die Ihnen auf dem Bildschirm dargeboten wird**, und finden einen Film, der Ihnen beiden gefallen könnte. **(10) Das INSPIRE-System soll Sie an den Beginn des Films erinnern**.

Es ist nun schon etwas dämmrig im Wohnraum geworden. Dem **(11) helfen Sie mit zwei Lichtern ab** und **(12) reduzieren die Helligkeit**. Später, als das System Sie an den Beginn des Filmes erinnert, lassen Sie den Abend wie geplant mit einem entspannenden Film ausklingen.

Szenario C

Situation:

Sie kommen zusammen mit Ihrem Partner von der Arbeit nach Hause. Es war ein anstrengender Tag mit vielen Sitzungen, und Sie hatten wenig Gelegenheit, über das Abendprogramm nachzudenken. Doch bevor Sie weiter planen, setzen Sie sich zunächst entspannt auf die Couch und **(1) bitten das INSPIRE-System, den Anrufbeantworter abzufragen**. Ein Freund hat angerufen und lädt Sie für den heutigen Abend zu einem Glas Sekt ein, um auf seinen Geburtstag anzustoßen. Eine zweite Nachricht betrifft eine neue Versicherung. Sie entschließen sich, **(2) beide Nachrichten zu sichern**.

Die Einladung erübrigt natürlich weitere Pläne für den Abend, und Ihr Partner ist ebenfalls von der Idee begeistert. Die Sonne des Tages hat das Wohnzimmer aufgeheizt und Sie **(3) lassen etwas das Rollo herunter** und **(4) sorgen mit dem Ventilator für Kühlung**.

Während Ihr Partner das Abendessen zubereitet, **(5) schalten Sie den Fernseher für die Abendnachrichten an**. Danach **(6) gehen Sie noch in die Programminformation** und informieren sich über das Abendprogramm. **(7) Beim Durchsehen der auf dem Bildschirm angezeigten Liste** bemerken Sie, dass heute abend eine weitere Folge der Serie "Friends" läuft. Sie **(8) bitten das System, diese Folge für später aufzuzeichnen**.

Es ist wieder angenehm kühl im Wohnraum geworden, und **(9) der Ventilator ist nicht mehr notwendig**. Sie **(10) sorgen mit zwei Lichtern für eine gute Beleuchtung**, **(11) aber reduzieren deren Helligkeit**. Bevor Sie gehen, **(12) bitten Sie das INSPIRE-System, alle Lichter zu löschen**, und Sie starten zu einem netten Geburtstagsabend.

7.2 INSPIRE Interaktionsfragebogen

Beurteilung der Interaktion

1. Gesamteindruck der Interaktion mit dem INSPIRE-System:

schlecht dürftig ordentlich gut ausgezeichnet

2. Erreichen der gewünschten Ziele:

2.1 Das System tat nicht immer das, was ich wollte.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

2.2 Die vom System gelieferten Informationen waren klar und deutlich.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

2.3 Die gelieferten Informationen waren unvollständig.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

2.4 Mit dem System lassen sich die Hausgeräte effizient bedienen.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

2.5 Das System ist unzuverlässig.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

3. Verständigung mit dem System:

3.1 Ich fühle mich gut vom System verstanden.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

3.2 Ich wusste zu jeder Zeit, was ich dem System sagen konnte.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

3.3 Ich musste mich konzentrieren, um das System akustisch zu verstehen.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

3.4 Die Stimme des Systems klang natürlich.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4. Verhalten des Systems:

4.1 Das System reagierte zu langsam.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.2 Das System ist freundlich.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.3 Das System reagierte nicht immer wie erwartet.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.4 Ich wusste nicht immer, was das System von mir verlangte.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.5 Das System machte viele Fehler.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.6 Ich konnte auftretende Fehler leicht beheben.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.7 Das System reagierte wie ein Mensch.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

4.8 Das System verhielt sich kooperativ.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

5. Gespräch:

5.1 Ich konnte leicht den Gesprächsfaden verlieren.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

5.2 Das Gespräch verlief holprig.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

5.3 Ich konnte das Gespräch wie gewünscht lenken.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

5.4 Das Gespräch war zu lang.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

5.5 Das Gespräch führte schnell zum gewünschten Ziel.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

5.6 Die Gesprächsanteile waren gleich verteilt zwischen mir und dem System.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

6. Persönliche Wirkung:

6.1 Die Interaktion mit dem System war angenehm.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

6.2 Ich fühlte mich entspannt.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

6.3 Ich musste mich sehr auf die Interaktion mit dem System konzentrieren.

stimme stark zu ☐ | stimme zu ☐ | unentschieden ☐ | lehne ab ☐ | lehne stark ab ☐

6.4 Die Interaktion hat Spaß gemacht.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

6.5 Ich bin insgesamt mit dem System zufrieden.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

*7. **Benutzbarkeit des Systems:***

7.1 Das System lässt sich nur schwer bedienen.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.2 Die Benutzung des Systems lässt sich leicht erlernen.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.3 Die Bedienung der Hausgeräte durch Sprache war komfortabel.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.4 Das System ist zu unflexibel.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.5 Das System ist nicht hilfreich zur Bedienung der Hausgeräte.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.6 Ich würde die Hausgeräte lieber auf andere Weise bedienen.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.7 Ich würde das System in Zukunft wieder benutzen.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.8 Die Benutzung des Systems hat sich gelohnt.

stimme stark zu ☐ stimme zu ☐ unentschieden ☐ lehne ab ☐ lehne stark ab ☐

7.3 Szenariobeschreibungen BoRIS

Dialogue no. ____

You would like to know where you can eat duck. Please ask BoRIS.

Restaurant name(s):

Dialogue no. ____

You plan to go out for a Greek dinner on Tuesday night in Grumme.

Price: -  +

Restaurant name(s):

If BoRIS is unable to indicate a restaurant, please change the following specification:

You want to have the dinner in Weitmar.

Restaurant name(s):

Dialogue no. ___

You plan to have your lunch break in a Chinese restaurant downtown.

Price: -  +

Restaurant name(s):

If BoRIS is unable to indicate a restaurant, please change the following specification:

The price.

Restaurant name(s):

Dialogue no. ____

Please gather your information from the following hints:

Price: - ├──┼──*──┼──┤ +

Type of food:

Location:

Restaurant name(s):

If BoRIS doesn't find a matching restaurant, please change the following:

Price: - ├──X──┼──┼──┤ +

Restaurant name(s):

Dialogue no. ___

You plan to eat out in Bochum. Because your favorite restaurant is closed for holidays, ask BoRIS for a restaurant.

Please write down first which specifications you want to give to BoRIS.

If BoRIS is unable to find a matching restaurant, please search for an alternative until BoRIS indicates at least one restaurant.

Restaurant name(s):

7.4 BoRIS Fragebogen (englische Version)

TS-No.:_____
Date:_____

BoRIS

Dear participant!

Thank you for taking your time for this experiment!

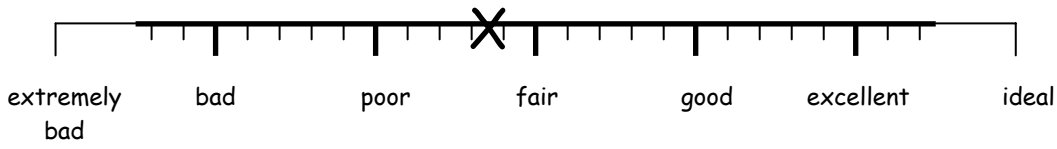
During the next hour you will get to know BoRIS via the telephone: The Bochum restaurant information system.

This test will show how you experience a telephone call with BoRIS. For this aim, we ask you to call BoRIS five times. Before each call you will get a small task. At the end of **each** telephone call, we ask you to write down what you think about the system. You can do this easily by filling out a questionnaire.

Before the test starts, we would like to ask you to answer the questions given on the following pages. For the test evaluation, we need some personal information from your side, information which will be treated anonymously of course.

At the end of the **whole** experiment, we ask you to give an overall judgment about all calls you had with BoRIS.

For some assessments you will find the following scale:



Usually, your judgment should be in the range between bad and excellent. In case of an unpredictable extreme judgment, you can use the thinly drawn edges of the scale as well. Please also use the spaces between the grid marks, as depicted above.

Assess the system in a very self-confident way and remember during the whole test session:

Not **you** are tested, but **you** test our system!

And now: Have a lot of fun!

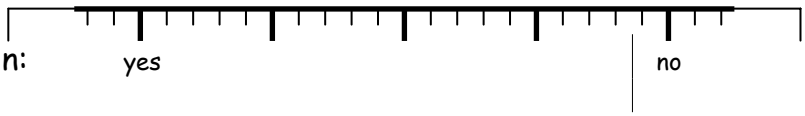
Part B

Overall impression:

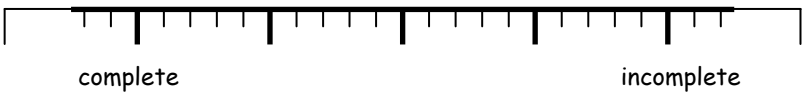
Extremely bad | bad | poor | fair | good | excellent | ideal

Information obtained from the system

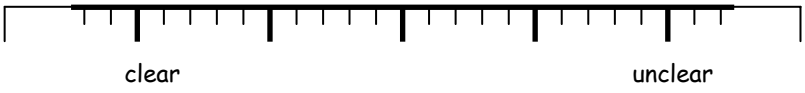
1. The system provided the desired information:



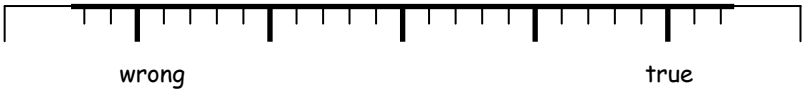
2. The provided information was...



3. The information was...

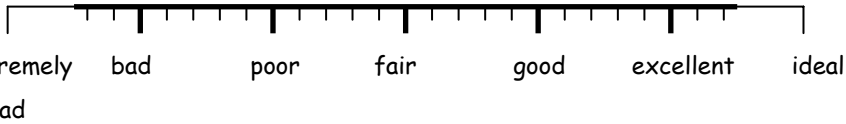


4. You would rate the information as...

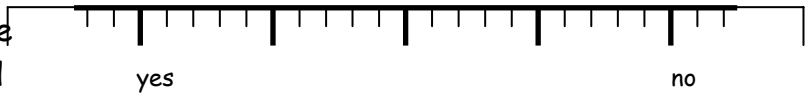


Communication with the system

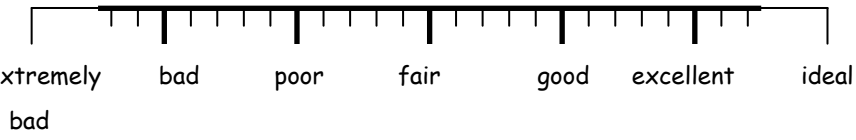
5. How well did you feel understood by the system?



6. You had to concentrate in order to understand what the system expected from you:



7. How well was the system acoustically intelligible?



System behavior

8. You knew at each point of the dialogue what the system expected from you:

9. In your opinion, the system processed your specifications correctly:

10. The system's behavior was always as you expected:

11. How often did the system make mistakes?

12. The system reacted in the same way as humans do:

13. The system reacted...

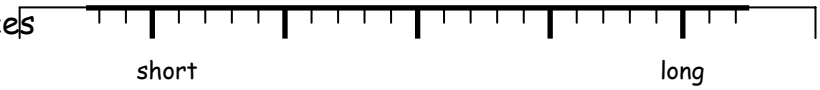
14. You were able to control the dialogue in the desired way:

15. The system reacted...

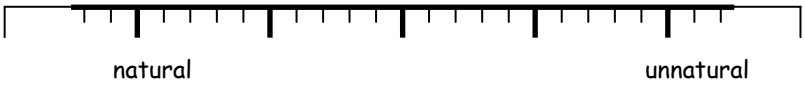
16. The system reacted in a ... way:

Dialogue

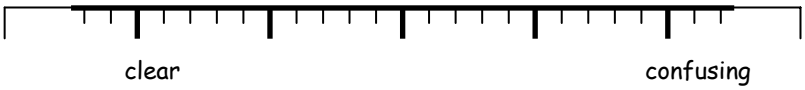
17. The system utterances were...



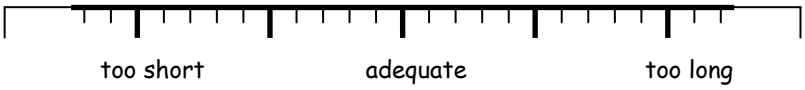
18. You perceived the dialogue as...



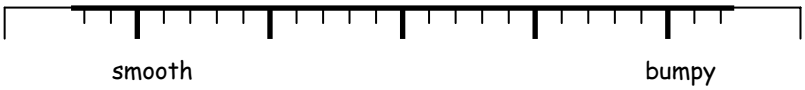
19. The course of the dialogue was...



20. The dialogue was...

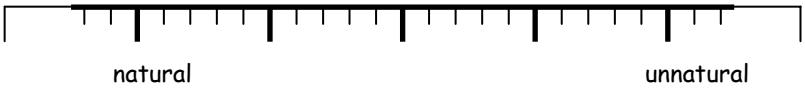


21. The course of the dialogue was...

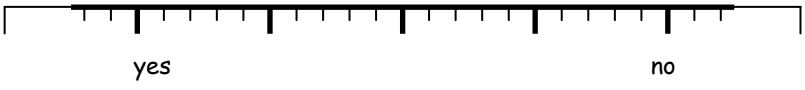


Your impression from the system

22. The system's voice was...

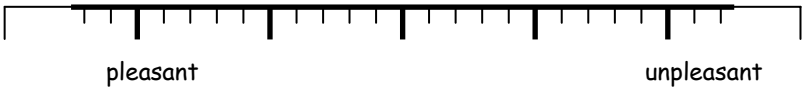


23. Overall, you are satisfied with the dialogue:

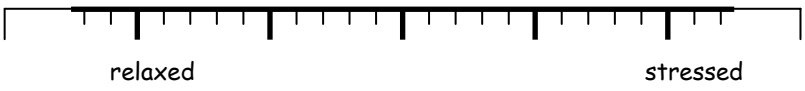


Personal impression

24. You perceived the dialogue as...



25. During the dialogue, you felt ...



7.5 Im Text referenzierte Ergebnisse der Berechnungen der Regressionsbäume

INSPIRE

Die folgende Auflistung der Ergebnisse ist wie folgt zu lesen: für jede Anzahl verbleibender Stufen (levelsleft) wurden pro Messgröße sieben Werte, einer für jede Prädiktorenkonfiguration gemessen. Der Schlüssel, welche Prädiktoren für den jeweiligen Wert zur Berechnung verwendet wurden, findet sich am Kopf der Liste. Die Zahlen unter „all_pred“ stellen die verwendeten Kombinationen der Prädiktoren dar. „Levelsleft“ gibt an, wie viele ebenen der Entscheidungsbaum nach dem Stutzen noch hatte. Die werte unter „all_correltest“ sind die Ergebnisse für die Berechnungen mit den verschiedenen Prädiktorkombinationen in der selben Reihenfolge wie deren Auflistung. Entsprechendes gilt für „all_rsquaretest“ (R^2) und „all_adjrsquaretest“ (adjusted R^2).

Inadequate Magnitude of Control (1)
Noun (2)
Space Modelling (3)
Reference (4)
Stagnation (5)
Repetition (6)
Help (7)

all_pred = 2 3 4 5 6 7
all_pred = 1 3 4 5 6 7
all_pred = 1 2 4 5 6 7
all_pred = 1 2 3 5 6 7
all_pred = 1 2 3 4 6 7
all_pred = 1 2 3 4 5 7
all_pred = 1 2 3 4 5 6

levelsleft = 1
all_correltest = -1 -1 -1 -1 -1 -1 -1
all_rsquaretest =-0.1142 -0.1142 -0.1142 -0.1142 -0.1142 -0.1142 -0.1142
all_adjrsquaretest =-0.6713 -0.6713 -0.6713 -0.6713 -0.6713 -0.6713 -0.6713

levelsleft = 2
all_correltest = 0.6908 0.6908 0.6908 0.6908 0.5305 **0.7596** 0.6908
all_rsquaretest = 0.4365 0.4365 0.4365 0.4365 0.2519 **0.5754** 0.4365
all_adjrsquaretest = 0.1547 0.1547 0.1547 0.1547 -0.1222 **0.3630** 0.1547

levelsleft = 3
all_correltest = 0.6573 0.6573 0.7311 0.7106 0.4391 0.7161 0.6573
all_rsquaretest = 0.3464 0.3464 0.4551 0.4470 0.0560 0.4866 0.3464
all_adjrsquaretest = 0.0196 0.0196 0.1827 0.1705 -0.4160 0.2299 0.0196

levelsleft = 4
all_correltest = 0.6642 0.6043 0.7040 0.6279 0.5228 0.7165 0.6497
all_rsquaretest = 0.3156 0.2031 0.4026 0.2583 0.1410 0.4473 0.3054
all_adjrsquaretest =-0.0265 -0.1954 0.1040 -0.1125 -0.2885 0.1710 -0.0418

BoRIS

External Restrictions (1)
Attribute Type (2)
Time Modelling (3)
Stagnation (4)
Repetition (5)
Regression (6)
Restart (7)

all_pred = 2 3 4 5 6 7
all_pred = 1 3 4 5 6 7
all_pred = 1 2 4 5 6 7
all_pred = 1 2 3 5 6 7
all_pred = 1 2 3 4 6 7
all_pred = 1 2 3 4 5 7
all_pred = 1 2 3 4 5 6

levelsleft = 1
all_correltest =-0.5987 -0.5987 -0.5987 -0.5987 -0.5987 -0.5987 -0.5987
all_rsquaretest =-0.0195 -0.0195 -0.0195 -0.0195 -0.0195 -0.0195 -0.0195
all_adjrsquaretest =-0.0553 -0.0553 -0.0553 -0.0553 -0.0553 -0.0553 -0.0553

levelsleft = 2
all_correltest = 0.2652 0.2652 0.2652 0.2652 0.2652 0.4529 0.4599
all_rsquaretest = 0.0347 0.0347 0.0347 0.0347 0.0347 0.2046 0.2108
all_adjrsquaretest = 0.0009 0.0009 0.0009 0.0009 0.0009 0.1767 0.1831

levelsleft = 3
all_correltest = **0.5266 0.5266 0.5266 0.5266 0.5266** 0.4856 0.4585
all_rsquaretest = **0.2756 0.2756 0.2756 0.2756 0.2756** 0.2332 0.2063
all_adjrsquaretest = **0.2502 0.2502 0.2502 0.2502 0.2502** 0.2063 0.1784

levelsleft = 4
all_correltest = 0.4994 0.4994 0.5000 0.5121 0.5025 0.4286 0.4710
all_rsquaretest = 0.2419 0.2419 0.2425 0.2570 0.2452 0.1653 0.2105
all_adjrsquaretest = 0.2153 0.2153 0.2160 0.2310 0.2187 0.1360 0.1828

levelsleft = 5
all_correltest = 0.4594 0.5049 0.4647 0.5132 0.4933 0.4274 0.4637
all_rsquaretest = 0.1762 0.2426 0.1956 0.2506 0.2251 0.1587 0.1981

all_adjrsquaretest = 0.1472 0.2160 0.1674 0.2243 0.1979 0.1292 0.1699

levelsleft = 6
all_correltest = 0.4505 0.5091 0.4390 0.4855 0.4463 0.4400 0.4032
all_rsquaretest = 0.1622 0.2410 0.1503 0.2055 0.1501 0.1638 0.1143
all_adjrsquaretest = 0.1328 0.2143 0.1205 0.1777 0.1203 0.1345 0.0832

levelsleft = 7
all_correltest = 0.4581 0.4810 0.4130 0.4719 0.4307 0.4351 0.3874
all_rsquaretest = 0.1688 0.1945 0.1006 0.1831 0.1142 0.1500 0.0748
all_adjrsquaretest = 0.1396 0.1662 0.0690 0.1544 0.0831 0.1202 0.0423

levelsleft = 8
all_correltest = 0.4298 0.4740 0.4109 0.4565 0.4300 0.4451 0.3491
all_rsquaretest = 0.1176 0.1824 0.0871 0.1597 0.1146 0.1515 0.0029
all_adjrsquaretest = 0.0866 0.1537 0.0551 0.1302 0.0835 0.1218 -0.0321

Literatur

Allen, Rober B., *Mental models and user models.* In: Helander, M., Landauer, T. K. und Prabhu, P., Handbook of Human-Computer Interaction. Second, completely revised edition. Elsevier Science, Amsterdam, Niederlande, 1997.

Boland, H., Hoonhout, J., Krebber, J., Möller, S., Schuchard, D., Melichar, M., Ganchev, T., Kladis, B., Smeele, P., *System Usability Evaluation Report*. Deliverable 6.2, IST project INSPIRE (INfotainment management with SPeech Interaction via REmote-microphones and telephone interfaces, IST-2001-32746), Institut für Kommunikationsakustik, Ruhr-Universität, Bochum, 2004.

Bortz, J., *Lehrbuch der Statistik. Für Sozialwissenschaftler*. Springer Verlag, Berlin u. a., 1977.

Bortz, J. und Lienert, G..A., *Kurzgefasste Statistik für die klinische Forschung. Leitfaden für die Verteilungsfreie Analyse kleiner Stichproben*. Springer Verlag, Berlin u. a., 2003.

Brandt, D. Scott und Uden, Lorna, *Insight into Mental Models Of Novice Internet Searchers*. In:Communicaions of the ACM, Vol. 46, Nr. 7, July 2003.

Compagnoni, B., *Development of Prediction Models for the Quality of Spoken Dialog Systems*. Diplomarbeit (unveröffentlicht), Deutsche Telekom Laboratories/Institut für Nachrichtentechnik, TU-Braunschweig, 2006.

Constantinides, P. C. und Rudnicky, A. I., *Dialog Analysis in the Carnegie Mellon Communicator*. In: Proceedings 6th European Conference on Speech Communication and Technology (EUROSPEECH ’99), Budapest, Ungarn, 1999, S. 243–246.

Gentner, D. und Gentner, D. *R*., *Flowing water of teeming crowds of electricity*. In: Gentner, D. und Stevens, A. (Hg), Mental Models, Erlbaum, Hillsdale (NJ), 1983, S. 99-129.

Hair, J. F., Anderson, *R*. E., Tatham, *R*. L., Black, W. C., *Multivariate Data Analysis (Fifth Edition)*. Prentice Hall, Upper Saddle River, NJ, 1998.

Hollnagel, E., *The Phenotypes of Erroneous Actions. Implications for HCI design*. In: Weir, G. *R*. S. und Alty, J. L. (Hg.), Human Computer Interaction and Complex Systems. Academic Press, London, 1989.

Johnson-Laird, P. N., Byrne, *R*. M. J. und Schaeken, W., *Propositional reasoning by model*. In: Psychological Review, 99, 1992, S. 418-439.

Jurafski, D.,Wooters, C., Tajchman, G., Segal, J., Stolcke, A., Fosler, E., and Morgan, N., *The Berkeley Restaurant Project*. In: Proc. 3rd Int. Conf. on Spoken Language Processing (ICSLP'94), Yokohama, Japan, 1994, 4:2139–2142.

Lang, K. L., Graesser, A. C. und Hemphill, D. D., *Understanding Errors in Human Computer Interaction*. In: SIGCHI Bulletin, Vol. 23, Nr. 4, Oktober 1991.

Moray, N., *Mental models in theory and practice*. In: Attention and performance. Cognitive regulation of performance. MIT press, Cambridge Mass. (u. a.), 1999.

Morris, N. M. und Rouse, W. B., *The effects of type of knowledge upon human problem solving in a process control task*. In: IEEE Transactions on Systems, Man, and Cybernetics, SMC-15, 1985, S. 698-707.

Möller, S., *Perceptual quality dimensions of spoken dialogue systems. A review and new experimental results*. In: Proceedings 4th European Congress on Acoustics (Forum Acusticum Budapest 2005), Budapest, Ungarn, 2005, S. 2681-2686.

Möller, S., *Quality of Telephone-Based Spoken Dialogue Systems*. Springer, New York, 2005(b).

Möller, S., Smeele, P., Boland, H., Krebber, J., *Evaluating Spoken Dialogue Systems According to De-Facto Standards: A Case Study*. In: Computer Speech and Language 21, 2007, S. 26-53.

Norman, D. A., *Categorization of action slips*. In: Psychological Review, 88, 1981, S. 1-15.

Norman, D. A., *Some observations on mental models*. In D. Gentner und A. L. Stevens (Eds.): Mental Models. S. 7-14, Erlbaum, Hilsdale (NJ), 1983, S. 7-14.

Oulasvirta, A., Möller, S., Engelbrecht, K.-P. and Jameson, A., *The Relationship of User Errors to Perceived Usability of a Spoken Dialogue System*. In: Proceedings 2nd ISCA/DEGA Tutorial and Research Workshop on Perceptual Quality of Systems, Berlin, 2006.

Prabhu, P. V. und Prabhu, G. V., *Human Error and User-Interface Design*. In: Helander, M., Landauer, T. K. und Prabhu, P., Handbook of Human-Computer Interaction. Second, completely revised edition. Elsevier Science, Amsterdam, Niederlande, 1997.

Rouse, William B. and Morris, Nancy M., *On looking into the black box. Prospects and limits in the search for mental models*. In: Psychological Bulletin, Vol. 100, No. 3, 1986, S. 349-363.

Sanderson, P. M., *Knowledge acquisition and fault diagnosis. Experiments with PLAULT*. In: IEEE Transactions on Systems, Man, and Cybernetics, SMC-20, 1990, S. 225-242.

Skowronek, J., *Entwicklung von Modellierungsansätzen zur Vorhersage der Dienstequalität bei der Interaktion mit einem natürlichsprachlichen Dialogsystem*.

Diplomarbeit (unveröffentlicht), Institut für Kommunikationsakustik, Ruhr-Universität Bochum, 2002.

Ergonomie der Mensch-System-Interaktion - Teil 110: Grundsätze der Dialoggestaltung (ISO 9241-110:2006); Deutsche Fassung EN ISO 9241-110:2006.