

Automatisierte Evaluierung von Mensch-Maschine-Schnittstellen

Stefan Schmidt¹ und Matthias Schulz¹

Schlüsselwörter: automatische Usability-Evaluierung, Simulation, Sprachdialogsystem, GUI

Zusammenfassung

Es werden fünf Werkzeuge zur automatischen Evaluierung von Mensch-Maschine-Schnittstellen (kurz „Interface“) vorgestellt. Bei den betrachteten Werkzeugen handelt es sich um CogTool, CogTool-Explorer, Biswas Simulator, MeMo sowie SpeechEval. Der Beitrag gibt Auskunft, welche Systeme mit welchem dieser Werkzeuge evaluiert sowie welche Resultate von der Evaluierung erwartet werden können. Den Abschluss bildet eine zusammenfassende Tabelle, welche die Eigenschaften und Merkmale der einzelnen Werkzeuge übersichtlich darstellt.

Abstract

Five tools for the evaluation of human-machine-interfaces are introduced. The examined tools are CogTool, CogTool-Explorer, Biswas' Simulator, MeMo and SpeechEval. The article will explain which tool is able to evaluate which kind of systems and what results one can expect from the simulation. A table, clearly showing each tool's properties and characteristics, completes the paper.

Einleitung

Die klassische Evaluierung eines Interfaces mit Hilfe einer geeigneten Gruppe von Versuchspersonen verursacht hohe Kosten, einen hohen organisatorischen Aufwand und erfolgt in den meisten Fällen zu spät, d.h. erst dann, wenn das Gesamtsystem und damit das Interface kurz vor der endgültigen Fertigstellung stehen. Techniken die es ermöglichen eine Evaluierung bereits während der Konzeptions- oder Entwicklungsphase, bspw. anhand von Mock-ups oder Prototypen, durchzuführen, unterstützen die Idee des Usability Engineering Lifecycle (Möller, 2010, S. 58.).

Der "Cognitive Walkthrough" (Nielsen, 1994, S. 105–140) reduziert den Aufwand von Usability-Untersuchungen und ist auch für die Evaluierung von Prototypen geeignet, allerdings dürfen auch hier die Kosten bei regelmäßiger Durchführung nicht außer Acht gelassen werden. Ebenso ist der Nutzen während der Konzeptionsphase eines Systems gering, da das zu testende System fehlt.

Werkzeuge zur semi-automatischen Usability-Evaluierung ermöglichen häufige und regelmäßige Untersuchungen von Systemen, auch dann, wenn sich diese noch in der Konzeption oder Entwicklung befinden. In diesem Beitrag geben wir eine Übersicht über fünf Anwendungen – namentlich CogTool, CogTool-Explorer, Biswas Simulator, MeMo Werkbank und SpeechEval – mit denen sich grafische, akustische und multimodale Schnittstellen automatisch evaluieren lassen.

Unser Ziel ist es einen Überblick darüber zu geben, welche Arten von Interfaces jeweils evaluiert werden können, welches Prinzip hinter dem jeweilige Modell des simulierten Nutzer steht sowie welche Art von Ergebnissen jeweils erwartet werden können. Für CogTool, CogTool-Explorer und MeMo Werkbank konnten wir, neben der Analyse von Publikationen,

¹ Technische Universität Berlin, Deutsche Telekom Laboratories, Quality and Usability Labs

eigene praktische Erfahrungen sammeln. An der Entwicklung der MeMo Werkbank sind die Autoren aktiv beteiligt.

Werkzeuge zur automatischen Usability-Evaluierung

CogTool

CogTool² (Bellamy, John & Kogan, 2011) ermöglicht die Modellierung von Systemen welche Tastatur, Maus, Touchscreen sowie Mikrophon (Sprache) als Eingabegeräte und Bildschirm sowie Lautsprecher als Ausgabegerät verwenden – wobei eine beliebige Kombination der Ein- und Ausgabegeräte möglich ist.

Das Systemmodell basiert auf einem endlichen Automaten, bei dem jeder Status einen möglichen Dialog darstellt. Jeder Dialog enthält die möglichen Elemente mit denen der Nutzer interagieren kann. So werden z.B. bei einem GUI Buttons, Auswahlboxen, Textfelder usw. anhand von Bildschirmfotos annotiert. Dadurch sind die Art der Dialogelemente und deren relative Lage zueinander festgelegt. Zu jedem Dialogelement kann eine Transition angegeben werden, welche zu einem Folgestatus führt, wenn das Element vom Nutzermodell verwendet wird (John, 2009). Möglich sind auch der Import kompletter Webseiten und die automatische Annotierung der enthaltenden Dialogelemente.

CogTool basiert zum Teil auf Ideen von GOMS (Card, Moran & Newell, 1983), weshalb kein spezielles Aufgabenmodell existiert, sondern der Designer die zur Lösung einer Aufgabe nötigen Schritte vorgibt (John, Prevas, Salvucci & Koedinger, 2004). Das Nutzermodell ist als Keystroke Level Model (KLM) realisiert. Für die Simulation werden die vorgegeben Lösungsschritte in atomare KLM Operatoren zerlegt und dann auf die entsprechenden ACT-Simple Kommandos abgebildet. Mit dem ACT-Simple Compiler (Salvucci & Lee, 2003) werden schließlich ACT-R³ (Anderson & Lebiere, 1998) Produktionsregeln erzeugt, mit denen dann die eigentliche Simulation durchgeführt wird. Diese Zerlegung, Übersetzung und Ausführung in ACT-R wird von CogTool durchgeführt und ist für den Nutzer nicht sichtbar.

CogTool Explorer

Der CogTool-Explorer (Teo & John, 2008) (CT-E), eine Erweiterung von CogTool, ermöglicht die Simulation der zielgerichteten Suche eines Nutzers, nach einem Link auf einer Webseite. Das Ziel des Nutzers wird durch einen Textstring beschrieben, der mittels einer semantischen Bewertung mit allen wahrgenommenen Links verglichen wird. Als Alternative zu ACT-R wird SNIF-ACT⁴ als kognitives Nutzermodell verwendet. SNIF-ACT beschreibt wie Nutzer, durch das Wahrnehmen und Bewerten von Links, auf Webseiten navigieren (Fu & Pirolli, 2007). Da allein dieses Suchen und Bewerten nicht ausreicht um die Ausführungszeit vorherzusagen, werden Operatoren für die visuelle Suche und Handbewegungen hinzugefügt, um genauere Vorhersagen zu erhalten.

Das Systemmodell wird automatisch mit der HTML-Import-Funktion von CogTool erstellt, und auch die Berechnung der semantischen Ähnlichkeit zwischen Nutzerziel und den einzelnen Links kann automatisch erfolgen.

Das, im Gegensatz zum ursprünglichen CogTool, nichtdeterministische CT-E Nutzermodell sucht selbstständig einen Lösungsweg zur Erreichung des Ziels, d.h. es werden verschiedene Lösungswege mit ihren jeweiligen Ausführungszeiten simuliert.

² Download von CogTool unter <http://cogtool.hcii.cs.cmu.edu> möglich.

³ ACT-R: Adaptive control of thought–rational

⁴ SNIF-ACT: Scent-based Navigation and Information Foraging in the ACT architecture

Das Resultat einer simulierten Interaktion ist sowohl in CogTool als auch CT-E die berechnete Interaktionszeit, sowie eine Visualisierung der einzelnen ACT-R Aktionen parallel zu dem erzeugten ACT-R Skript.

Simulator von Biswas

Biswas & Robinson, 2008 beschreiben einen Simulator zur Evaluierung von GUI für Systeme, welche von Menschen ohne Einschränkungen sowie mit Seh- oder Mobilitätseinschränkungen genutzt werden können. Die Validität der Ergebnisse wird in (Biswas & Robinson, 2010) gezeigt. Mit dem Simulator kann die Interaktion mit einem GUI über ein Cursorbasiertes Eingabegerät (z.B. Maus) simuliert werden.

Wie in CogTool basiert das Systemmodell auf einem endlichen Automaten mit einem Dialog (Bildschirm) je Status und Transitionen als Aktionen. Für die Aktionen werden keine konkreten Elemente angegeben, sondern nur die Positionen auf dem Bildschirmfoto.

Das Benutzermodell setzt sich aus drei einzelnen Modellen (perzeptives, kognitives und motorisches Modell) zusammen und kann mit Angaben zu Art und Schwere der Mobilitätseinschränkung der simulierten Benutzergruppe konfiguriert werden.

Als Erstes wird mit Hilfe des Modells der visuellen Wahrnehmung die Suche nach der nächsten zu nutzenden Interaktionsmöglichkeit simuliert. Dieses Modell beruht auf der Analyse von Bitmaps und kann durch entsprechende Verfremdung der Bildschirmfotos visuelle Einschränkungen berücksichtigen.

Das kognitive Modell entscheidet ob die gewünschte Aktion gefunden wurde. Eine GOMS basierte Implementierung ahmt Expertenverhalten nach, während ein mit empirischen Daten trainierbares Modell, die Entscheidungen von Novizen simuliert. Im ersten Fall gibt der Designer den Lösungsweg vor, im Zweiten ist ein vorhergehendes Experiment mit Versuchspersonen notwendig. Die dabei erhobenen Verläufe können auch für die automatische Evaluierung geänderter Versionen einer Anwendung verwendet werden.

Schließlich macht ein statistisches Modell des motorischen Verhaltens Aussagen über die Verläufe der Cursorbewegungen. Dieses Modell wurde mit Daten aus einer Studie von Trewin & Pain (1999) erstellt.

Für jede simulierte Interaktion liefert der Simulator Angaben zur Interaktionsdauer, den Bewegungen des Cursors und den Augenbewegungen bei der Betrachtung des Bildschirms.

MeMo Werkbank

Mit der MeMo Werkbank (MMW) ist es möglich Interaktionen mit GUIs und Sprachdialogsystemen (SDS) zu simulieren (Engelbrecht, Kruppa, Möller & Quade, 2008). Um die Modellierung beliebiger Systeme zu ermöglichen wird auch hier ein endlicher Automat als Basis für das Systemmodell verwendet.

Ähnlich wie bei CogTool werden für graphische Interfaces die Interaktionselemente anhand von Bildschirmfotos annotiert. Im Gegensatz zu CogTool und dem Biswas Simulator können je Dialogelement mehrere Transitionen (also möglich Aktionen) angelegt werden. Über Variablen, welche vor Ausführung einer Aktion geprüft werden (Bedingungen) und die nach der Ausführung verändert werden können (Konsequenzen) kann das Verhalten des Systemmodells während der Simulation geändert werden. Dies ermöglicht es auch Teile der Programmlogik des zu evaluierenden Systems zu modellieren. Der Austausch von Informationen erfolgt teils auf Textebene und teils auf Konzeptebene, letzteres insbesondere bei der Evaluierung eines SDS.

In der MMW wird ein regelbasiertes, probabilistisches Benutzermodell verwendet. Die dem Benutzermodell zu Verfügung stehenden Aktionen werden mit Wahrscheinlichkeitswerten ge-

wichtet, welche durch Charakteristika des Interfaces und des Nutzers (bspw. Alter, Sehkraft, technische Fähigkeiten, Aufmerksamkeit) beeinflusst werden. Der Grad des Einflusses ist durch Regeln abgebildet, die die Charakteristika auswerten (Schmidt, et al., 2010). Wie bei Biswas, orientiert sich die Arbeitsweise des Nutzermodells an dem Prinzip des „Human Processor“ nach Card, Moran & Newell (1986), mit der Ausführungskette Wahrnehmung, Verarbeitung und motorische Umsetzung.

Die Simulation der MMW liefert eine Visualisierung der Interaktionsverläufe (einzeln und kumuliert), die die durchlaufenden Status darstellt und farbliche Hervorhebung bei Abweichung vom kürzest möglichen Lösungspfad einfügt. Für Abweichungen kann abgefragt werden, aufgrund welcher Regel(n) das Nutzermodell eine Aktion ausgeführt bzw. nicht ausgeführt hat. Die Zuordnung einer Regel zur Abweichung ermöglicht ein Rückschluss auf eventuelle Usability-Probleme.

Ebenso können Berichte erzeugt werden, welche detailliert den Zustand des System- und Nutzermodells in jedem Status widerspiegeln und summative Informationen zu Aufgabenerfolg und Ausführungszeit geben. Speziell für SDS werden alle Parameter ausgegeben die nötig sind um eine Vorhersage der Nutzerzufriedenheit mittels PARADISE (Walker, Litman, Kamm & Abella, 1997) durchzuführen.

SpeechEval

Ausschließlich für die Evaluierung von Sprachdialogsystemen ist SpeechEval (Scheffler, Roller & Reithinger, 2009) entwickelt worden. Die Besonderheit an SpeechEval (SE) ist, dass das Nutzermodell per Telefon (oder eine andere Audioverbindung) direkt mit einem realen SDS interagiert. SpeechEval ist also vor allem dazu geeignet, ohne Nutzertests durchführen zu müssen, einen Korpus mit realistischen Dialogen zu erzeugen, welcher anschließend weiter ausgewertet werden kann.

In SpeechEval verfügt das Nutzermodell über eine Sprecherkennung (ASR) und ein Modul zur Sprachsynthese (TTS). Während der Simulation, die auf Signalebene stattfindet, übersetzt die ASR das Sprachsignal des SDS in Text. Nach dessen Auswertung erzeugt das Nutzermodell, auf Basis eines statistischen Modells, eine Äußerung, welche mit Hilfe der TTS wiederum als Sprachsignal zu dem SDS gesendet wird. Da die Kommunikation bereits über einen realen Kanal mit realen Störungen stattfindet, werden keine künstlichen Fehler in der ASR oder TTS des Nutzermodells erzeugt.

Das Modell zur Erzeugung der Nutzeräußerungen beruht auf einem, im Rahmen des SpeechEval Projektes annotierten, Korpus von Interaktionen aus dem VoiceAward (ein Wettbewerb für kommerzielle SDS) aus den Jahren 2005 bis 2008 (1900 Dialoge, 130 verschiedene SDS). Ziel der Verwendung eines solch großen Korpus war es, das Nutzermodell mit allgemeinen Strategien zu trainieren, welche auch bei der Interaktion mit neuen SDS realistische Dialoge erzeugen. SpeechEval liefert auch Werkzeuge, um das Nutzermodell mit einem eigenen domänenspezifischen Korpus, zu trainieren.

Die Simulationen liefern als Ergebnis Logdateien welche den künstlich erzeugten Dialog beschreiben. Nach Möller, et al. (2009) können diese Logs genutzt werden um mittels Hidden Markov Models Vorhersagen über die Bewertung des SDS durch reale Nutzer zu treffen.

Zusammenfassung

Einen kompakte Zusammenfassung der oben beschriebenen Eigenschaften gibt Tabelle. 1. Da in CogTool der Weg durch den Designer vorgegeben wird, entfällt eine selbständige Erkennung von und Entscheidung für Interaktionselemente. Aufgrund der gelieferten Art der Resultate sind CogTool, CogTool-Explorer und der Biswas Simulator eher dazu geeignet zwei oder mehr Systeme bzw. Systemvarianten miteinander zu vergleichen, während die MeMo Werkbank und

SpeechEval Daten liefern, auf deren Grundlage Ursachen für Usability-Probleme gesucht werden können.

Tab. 1: Eigenschaften der vorgestellten Werkzeuge. Legende: G: Grafisch, S: Sprache, T: Touch, vgg.: vorgegeben, -: trifft nicht zu.

	CogTool	CogTool-Explorer	Biswas Simulator	MeMo Werkbank	SpeechEval
Benutzermodell					
Wahrnehmung	vgg.	SNIF- ACT	Biswas	probabilis- tisch, regelbasiert	-
Kognition	vgg. / ACT-R	SNIF- ACT	GOMS, statistisch		statistisch
Motorik	ACT-R	ACT-R	statistisch		-
Training möglich	nein	nein	ja	nein	ja
Simulationsebene	Text	Text	Konzept	GUI: Text + Konzept, SDS: Konzept	Signalebene
Systemmodell					
Art	Modell	Modell	Modell	Modell	reales Sys.
Dynamik	statisch	statisch	statisch	dynamisch	dynamisch
Interaktions- elemente	Standard- widgets	Hyper- links	Ausschnitte	Standard- widgets	Sprache
Eingabemodali- täten	G, S, T	G	G	G, S	S
Eingabegeräte	Tastatur, Maus, Touch- screen, Mikrofon	Maus	Cursorba- siert	Maus, Mikrofon	Mikrofon, Telefon, usw. (Systemab- hängig)
Ausgabemodali- täten	G, S	G	G	G, S	S
Art der Evaluierungsergebnisse					
Grafisch	Dauer + Visualisie- rung		Dauer, Cursorspur, Augenbewe- gung	Visualisierung, detailliertes Log, Dauer	-
Sprache		-	-	Konzeptfehler, in PARADISE verw. Param.	Log der Kommuni- kation
Touch		-	-	-	-

Ausblick und Danksagung

Aufgrund der Fähigkeit von CogTool und der MeMo Werkbank multimodale Systeme zu evaluieren soll in einer nachfolgenden Arbeit die Güte der Vorhersagen, der Aufwand der Modellierung und mögliche Synergieeffekte bei gemeinsamer Nutzung näher untersucht werden.

Die Arbeit entstand im Rahmen des durch die Deutsche Forschungsgemeinschaft geförderten Projektes „UserModel“ (Fördernummer: MO 1038/8-1).

Literatur

- Anderson, J. R. & Lebiere, C. J. (1998). *The Atomic Components of Thought*: Lawrence Erlbaum Associates Inc.
- Bellamy, R. K. E., John, B. E. & Kogan, S. (2011). *Deploying CogTool: integrating quantitative usability assessment into real-world software development*. ICSE 2011 (S. 691–700).
- Biswas, P. & Robinson, P. (2010). *Evaluating the design of inclusive interfaces by simulation*. Proc. of IUI 2010 (S. 277–280). Erweiterte Fassung des Artikels online verfügbar: <http://www.cl.cam.ac.uk/~pr10/publications/iui10.pdf>.
- Biswas, P. & Robinson, P. (2008). *Automatic evaluation of assistive interfaces*. IUI 2008: (S. 247–256).
- Card, S. K., Moran, T. P. & Newell, A. (1986). *The model human processor: an engineering model for human performance*. In Handbook of Perception and Human Performance. (Band 2, 1–35). New York: Wiley.
- Card, S. K., Moran, T. P. & Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Erlbaum.
- Engelbrecht, K.-P., Kruppa, M., Möller, S. & Quade, M. (2008). *MeMo workbench for semi-automated usability testing*. Proc. of Interspeech 2008 (S. 1662–1665).
- Fu, W.-T. & Pirolli, P. (2007). *SNIF-ACT: A Cognitive Model of User Navigation on the World Wide Web*. Human-Computer Interaction, 22 (4), S. 355–412
- John, B. E. (2009). *CogTool User Guide*. Verfügbar unter: http://cogtool.hcii.cs.cmu.edu/sites/default/files/CogToolUserGuide_1_1.pdf.
- John, B. E., Prevas, K. C., Salvucci, D. D. & Koedinger, K. R. (2004). *Predictive human performance modeling made easy*. Proc. Of CHI 2004 (S. 455–462).
- Möller, S. (2010). *Quality Engineering. Qualität kommunikationstechnischer Systeme*. Heidelberg, Germany: Springer.
- Möller, S., Schleicher, R., Butenkov, D., Engelbrecht, K.-P., Gödde, F., Scheffler, T., Roller, R. & Reithinger, N. (2009). *Usability Engineering for Spoken Dialog Systems via Statistical User Models*. Proc. of IWSDS 2009.
- Nielsen, J. (1994). *Usability inspection methods*. New York: Wiley.
- Salvucci, D. D. & Lee, F. J. (2003). *Simple cognitive modeling in a complex cognitive architecture*. Proc. of SIGCHI 2003: (S. 265–272).
- Scheffler, T., Roller, R. & Reithinger, N. (2009). *SpeechEval - Evaluating Spoken Dialog Systems by User Simulation*. Proc. of IJCAI 2009 (S. 93–98).
- Schmidt, S., Engelbrecht, K.-P., Schulz, M., Meister, M., Stubbe, J., Töppel, M. & Möller, S. (2010). *Identification of interactivity sequences in interactions with spoken dialog systems*. PQS 2010 (S. 109–114).
- Teo, L. & John, B. (2008). Towards a tool for predicting goal-directed exploratory behavior. *Human Factors and Ergonomics Society Annual Meeting Proceedings* (S. 950–954).
- Trewin, S. & Pain, H. (1999). *Keyboard and mouse errors due to motor disabilities*. International Journal of Human-computer Studies, 50, (S. 109–144).
- Walker, M. A., Litman, D. J., Kamm, C. A. & Abella, A. (1997). *PARADISE: a framework for evaluating spoken dialogue agents*. Proc. of EAACL 1997 (S. 271–280).