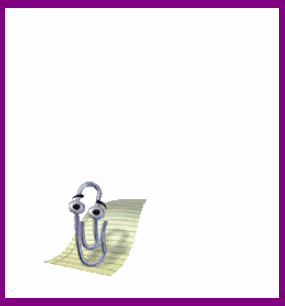
A User Acceptance Equation for Intelligent Assistants

Brad A. Myers
Human-Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
http://www.cs.cmu.edu/~bam
bam@cs.cmu.edu

AAAI 2007 Spring Symposium on Interaction Challenges for Intelligent Assistants

How To Make Users Happy

- And avoid annoying users



User Happiness?

$H_u = f(\text{Performance})$

User Happiness?

$H_u = f(\text{Performance, Trust})$

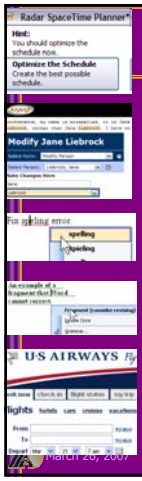
User Happiness!

$H_u = f(E_{Assistant}\ E_{Negative}\ E_{Positive}\ E_{Value}\ E_{User}\ E_{Corrected}\ E_{By\text{-}hand}\ E_{Cost}\ E_{Avoided}\ E_{Apparentness}\ E_{Correct\text{-}difficulty}\ E_{Sensible}\ W_{Quality}\ W_{Commitment}\ T_{By\text{-}hand}\ T_{By\text{-}Hand\text{-}start\text{-}up}\ T_{By\text{-}Hand\text{-}per\text{-}unit}\ T_{Assistant}\ T_{Training\text{-}start\text{-}up}\ T_{Assistant\text{-}per\text{-}unit}\ T_{Interaction\text{-}per\text{-}unit}\ T_{Monitoring}\ T_{Correcting}\ T_{Responsiveness}\ T_{System\text{-}Training}\ T_{User\text{-}training}\ T_{Average\text{-}for\text{-}each\text{-}correction}\ A_{Error\text{-}rate}\ N_{units}\ P_{Pleasantness}\ U_{Perceive}\ U_{Why}\ U_{Provenance}\ U_{Predictability}\ I_{Assistant\text{-}interfere}\ I_{Screen\text{-}space}\ I_{Cognitive}\ I_{Appropriate\text{-}Time}\ C_{Autonomy}\ C_{Correcting}\ S_{Sensible\text{-}Actions}\ S_{User\text{-}models}\ S_{Learning}\ R_{Social\text{-}Presence}\ D_{Hand}\ V_{Importance})$

Why Happiness?

- Focus on assistants that take on tasks which the users could do themselves
- Assistants are supposed to be helpful
- If not, users can turn off the assistants
  - Optional
  - Assume: cannot require users to use assistant or to provide feedback
- So only used if user finds it:
  - Useful
  - Trustable
  - Usable

Adjustable Autonomy



- Assistant does it all; completely autonomous
- User monitors actions of assistant (confirmation of assistant's actions)
- Assistant helps user do actions (or user tells agent how to do the actions)
- Assistant tells users where actions might be done
- User does all actions; direct manipulation

Rajiv T. Maheswaran, Milind Tambe, Pradeep Varakantham, Karen Myers, "Adjustable Autonomy Challenges in Personal Assistant Agents: A Position Paper", *Agents and Computational Autonomy*, Springer, 2004, pp. 187-194.

Key Factors

- Correctness
  - Errors
- Speed
  - Time to use system with the assistant
- Pleasantness
- Utility

Measures for Correctness

- Can measure % correct on corpus
- Or measure in field deployment
  - Often performance is much worse
- Also important is:
  - Overhead of monitoring for correctness
  - Time for correction
- If Assistant can be wrong, user might need to check each action
- When *is* wrong, need to:
  - Notice is wrong
  - Fix the error
- How long does this take compared to just doing it?
- But doing it by hand might have errors too!

Correctness

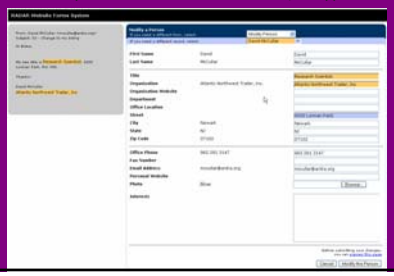
- How many errors does the assistant make?
  - $E_{Negative}$ False negatives: missed opportunities to help ("coverage")
    - Just silent when might do something
  - $E_{Positive}$ False positives: incorrectly offered to help ("precision")
  - E_{Value} Wrong values: partially correct, but with inaccurate parameters
- Total errors left in the results
  - E_{User} User's errors also involved
  - $E_{Corrected}$ User might catch errors and fix them

$$E_{Assistant} = E_{User} + E_{Positive} + E_{Value} - E_{Corrected}$$

- $E_{By\text{-}hand}$ But compare to errors when no assistant
- Error rate may change over time, as the assistant learns

Examples

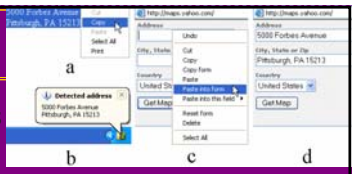
- Radar VIO (Virtual Information Officer) helps fill in form fields from emails
- John Zimmerman, Anthony Tomasic, Isaac Simmons, Ian Hargraves, Ken Mohnkern, Jason Cornwell, Robert Martin McGuire, "VIO: a mixed-initiative approach to learning and automating procedural update tasks". *CHI'2007 conference*, To appear.



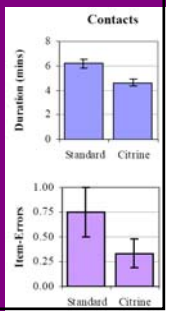
VIO Error Rate

- With VIO: Overall decrease in time by 17% ($p < .001$)
- Overall error rate (all users)
  $E_{Assistant}$ 15 (total errors left in result) vs.
  $E_{By\text{-}hand}$ 12 (*n.s.*)
- Per user error rates (20 users):
  $E_{Negative}$ 12 (missed extracting values)
  $E_{Positive}$ 0
  E_{Value} 1
- VIO strong biased away from incorrect guesses, so prefers not to say anything (favors $E_{Negative}$)

Example: Citrine Errors

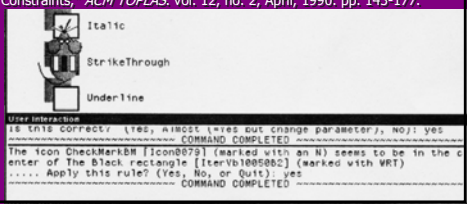


- Interprets addresses in copied text
- Copy-and-paste by hand for people's addresses took more time even including fixing errors, compared to using the Citrine assistant
- When by hand: left more errors in result
  - Jeffrey Stylos, Brad A. Myers, Andrew Faulring, "Citrine: Providing Intelligent Copy-and-Paste." *ACM Symposium on User Interface Software and Technology, UIST'04*, October 24-27, 2004, Santa Fe, NM. pp. 185-188.



Old Example

- Péridot (1985); confirm by question and answer
- Low consequence of errors
- Users generally just said "Yes" without understanding the question
  - Assumed computer knew better than they did
- So can't necessarily trust user's feedback
- Brad A. Myers. "Creating User Interfaces Using Programming-by-Example, Visual Programming, and Constraints," *ACM TOPLAS*. vol. 12, no. 2, April, 1990. pp. 143-177.



Consequences of Errors

- Not just a factor of the time for errors
- Other factors:
  - E_{Cost} Cost (seriousness) of making an error
    - Probably a key factor in user's acceptance and happiness
    - Aircraft auto-pilot vs. filling in addresses for a contact
  - Likelihood of making an error by hand compared to by the assistant (error avoidance):
    $E_{Avoided} = E_{By\text{-}hand} - E_{Assistant}$
  - $E_{Apparentness}$ Likelihood of noticing an error
  - $E_{Correct\text{-}difficulty}$ Ease of correction of the error
    - Likelihood of being able to correct it after finding it
    - Is the right information available?

Quality of Errors

- User happiness may not *only* depend on frequency and severity of errors
- Henry Lieberman: Depends on whether the errors make sense
  - Predictable vs. seemingly random errors
  - Knowledge-based vs. statistical techniques
  - But often errors easier to notice if very far off
    - Example: OCR, mistakes in Citrine
- Helps users predict how to avoid errors
- $E_{Sensible}$

Quality of Work Beyond Errors

- May not be right vs. wrong
- Quality of the assistant's work
  - Mary Shaw: Satisfactory level of work
    - E.g., Meeting transcripts
  - $W_{Quality}$
- Wayne Iba
  - Bad answers may inspire user to better work
    - Apprentice
  - $W_{Commitment}$ User's attitude and commitment affects quality

Measuring Time

- Time when performing tasks
- Can measure the time for the user without assistant, compared to with assistant
- Can include time to correct errors
  - But only those that the user notices
  - Corrected errors vs. Un-corrected errors
- Usually want the time to be faster when using the assistant
- May be slower for 1st time, but faster if used a lot
  - Because of training, learning time, etc.
- Does not include "background" time
  - Assistant can work in parallel to user

Equations for Time

- Control condition:

$$T_{By\text{-}hand} = T_{By\text{-}Hand\text{-}start\text{-}up} + (T_{By\text{-}Hand\text{-}per\text{-}unit} * N_{units})$$

- Time with assistant, including errors:

$$T_{Assistant} = T_{Training\text{-}start\text{-}up} + (T_{Assistant\text{-}per\text{-}unit} * N_{units})$$

- Where:

$$T_{Assistant\text{-}per\text{-}unit} = T_{Interaction\text{-}per\text{-}unit} + T_{Monitoring} + T_{Correcting} + T_{Responsiveness}$$

$$T_{Correcting} = A_{Error\text{-}rate} * T_{Average\text{-}for\text{-}each\text{-}correction}$$

$$A_{Error\_rate} = (E_{Positive} + E_{Value}) / T$$

Time per Item

- $T_{Interaction\text{-}per\text{-}unit}$ is average across items
  - Ones handled correctly by assistant lowers average time
  - If agent *anticipates* and does task, then small or 0
- Should be lower than $T_{By\text{-}Hand\text{-}per\text{-}unit}$ or will never win
- $T_{Responsiveness}$ Includes time that user has to wait for assistant
  - If agent slows down interaction
  - Also, if agent is slow, makes it look stupid
  - People don't like to wait even if overall is faster
    - Xerox Star judged poorly even though overall faster tasks
  - Conversely, people feel fast when busy with DM

Time and Accuracy

- What accuracy rate is required?

$T_{By\text{-}hand} \geq T_{Assistant} = T_{Training\text{-}start\text{-}up} + ((T_{Interaction\text{-}per\text{-}unit} + T_{Monitoring} + (A_{Error\text{-}rate} * T_{Average\text{-}for\text{-}each\text{-}correction})) * N_{units})$

- This formula can help determine how much accuracy is required for assistant to be worthwhile
- Can improve performance by improving UI for monitoring and correcting!
- If importance of checking is low, then user might not check any/all

Issues with Training Time

- $T_{Training\text{-}start\text{-}up}$ includes system training and user training

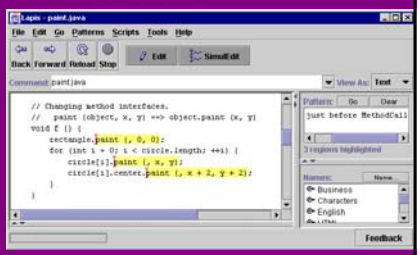
$T_{Training\text{-}start\text{-}up} = T_{System\text{-}training} + T_{User\text{-}training}$

$T_{By\text{-}hand} \geq T_{Assistant} = T_{Training\text{-}start\text{-}up} + ((T_{Interaction\text{-}per\text{-}unit} + T_{Monitoring} + (A_{Error\text{-}rate} * T_{Average\text{-}for\text{-}each\text{-}correction})) * N_{units})$

- $T_{System\text{-}training}$ is explicit training requires
  - Might be labeling examples, entering rules, specifying policies & permissions, etc.
  - CALO users complain about re-training required with every new release
  - Other assistants "pre-train" on corpus or do not need training, so: $T_{System\text{-}Training} = 0$
  - Try to get training from what users do anyway (implicit) so no extra overhead
- $T_{User\text{-}training}$ is time for user to learn how to use the assistant

Example of Performance Measures

- When there are repeated tasks, can measure cross-over point (N_{units})
  - When there are enough tasks to overcome the overhead
- Example, LAPIS supported "simultaneous editing"
  - Teach a pattern and edit all locations at once
  - Robert C. Miller and Brad A. Myers. "Interactive Simultaneous Editing of Multiple Text Regions." *USENIX 2001 Annual Technical Conference*, Boston, MA, June 2001. pp. 161-174.



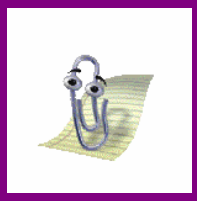
Example of Performance Measures

- Measure when enough tasks to overcome the overhead

Task	Records in task	Simultaneous editing	Manual editing	Equivalent task size: novices	Equivalent task size: expert
1	9	142.9 s [63-236 s]	21.6 s/rec[7.7-65 s/rec]	8.4 recs [2.1-12.2 recs]	4.5 recs
2	7	119.1 s [64-209 s]	32.3 s/rec [19-40 s/rec]	3.6 recs [1.9-5.8 recs]	1.6 recs
3	7	159.6 s [84-370 s]	41.3 s/rec [16-77 s/rec]	4.0 recs [1.9-6.2 recs]	2.4 recs

Subjective Factors

- How much do users like the assistant?
- Can be annoying even when not doing anything
- Alternatively, might be considered positively
  - Cute, helpful, polite, …
- $P_{Pleasantness}$



Factors of Pleasantness

- Understandability
- Interference
  - Interruptions
- User control
- Sensible Help
- Social Presence

$P_{Pleasantness} = F(U_{Perceive}\ U_{Why}\ U_{Provenance}\ U_{Predictability}\ I_{Assistant\text{-}interfere}\ I_{Screen\text{-}space}\ I_{Cognitive}\ I_{Appropriate\text{-}Time}\ C_{Autonomy}\ C_{Correcting}\ S_{Sensible\text{-}Actions}\ S_{User\text{-}models}\ S_{Learning}\ R_{Social\text{-}Presence})$

Understandability

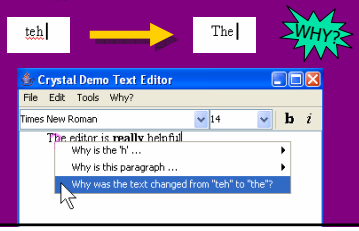
- Does user understand what is happening?
- Related user interface principles (Nielsen's Heuristics):
  - http://www.useit.com/papers/heuristic/heuristic_list.html
  - Visibility of system status, Recognition rather than recall, Aesthetic and minimalist design, Help users recognize, diagnose, and recover from errors, Help and documentation
- User able to perceive what the system is doing
  - $U_{Perceive}$ Actions, states, reasons are visible
- Understand why actions are being taken ("Transparency")
  - U_{Why} Lots of work on this topic
  - And understand the assistant's answers
- Interacts with control
  - Not just *understand* why
  - Also, be able to *change or fix it*
    - Not do it the same way next time

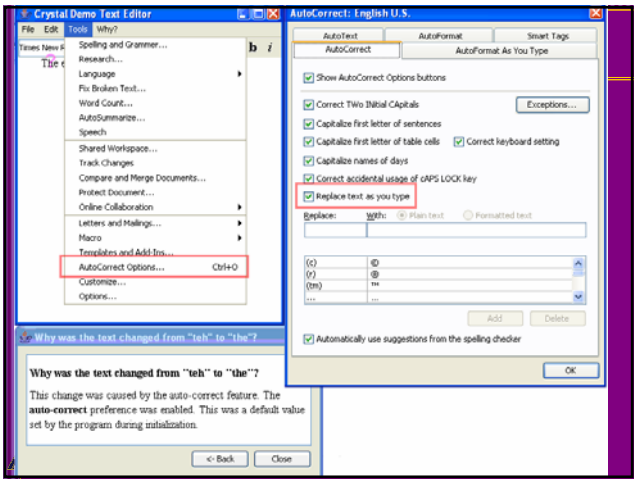
Understanding Values

- Understand actions assistant does
- Also important: understanding values
  - Where the values come from
  - Conley & McGuinness: Provenance and Credibility of values
  - $U_{Provenance}$

Explaining Why

- Crystal – my system to explain "why" for assistants in complex applications like Microsoft Word
- Doesn't just explain why, but brings up the dialog boxes to let user change it
- Brad Myers, David A. Weitzman, Andrew J. Ko, and Duen Horng Chau, "Answering Why and Why Not Questions in User Interfaces," *Proceedings CHI'2006: Human Factors in Computing Systems*. Montreal, Canada, April 22-27, 2006. pp. 397-406.





Predictability

- Can the user *predict* what the agent will do?
- Related user interface principles:
  - Consistency, Visibility of system status, Match between system and the real world
- Predictable <-> understandability
  - Understand future actions
  - Not just what it has already done
- $U_{Predictability}$

Interference

- $I_{Assistant\text{-}interfere}$ How much does the assistant interfere with other tasks?
  - Can make user less effective on unrelated tasks
- $I_{Screen\text{-}space}$ Screen space for the assistant
  - Compare Clippy vs. squiggly underlines
  - Towel's To-Do list window; TamaCoach's GUI
  - Really big explanations (Crystal, CALO, etc.)
  - Radar repeats email with the assistant's interpretation

Interference

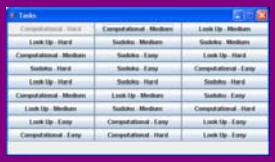
- $I_{Cognitive}$ Cognitive overhead of monitoring assistant
  - Attention taken away from other tasks
  - Example: Meeting Rapporteur mentions checking/correcting assistant's notes compared to participating in meeting
    - Vs. taking notes by hand
- $T_{Monitoring}$ Time overhead already included

Interruptions

- Interruptions interfere
- Can be annoying
- But may be necessary
  - E.g., RoboCare notification for medicine
- Some systems trying to predict appropriate times to interrupt
- Decisions:
  - Whether to interrupt
    - Vs. perform autonomously or not assist at all
  - How to ask the question (understandability)
  - When to interrupt $I_{Appropriate-Time}$

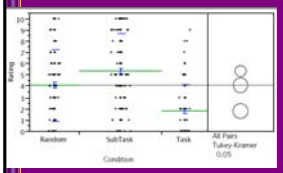
Interruptions, example

- Radar Attention Manager
  - Dan Siewiorek & Asim Smailagic



Radar Attention Manager

- Subject rating of interruption annoyance (1 Low, 10 High)
  - Subtask boundaries *worse* than random; tasks boundaries better
- Good success at predicting when interruptible



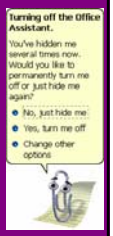
Classifier	Accuracy	True positives	True negatives
PSVM	**85.35 %**	82.69 %	88.00 %
LSVM	81.24 %	80.77 %	81.71 %
LDA	75.47 %	69.23 %	81.71 %
MNN	66.74 %	55.77 %	77.71 %

User Control

- Ability of user to control the assistant
- Related user interface principles:
  - User control and freedom, Error prevention, Flexibility and efficiency of use, Help users recover from errors
- $C_{Autonomy}$ Control the level of autonomy
  - Related to assistant overhead: $T_{Training\text{-}start\text{-}up}$
- $C_{Correcting}$ Difficulty of fixing results of errors
  - Related to $T_{Monitoring} + T_{Correcting}$
  - Also possibly mental difficulty of doing this process
    - Not just time

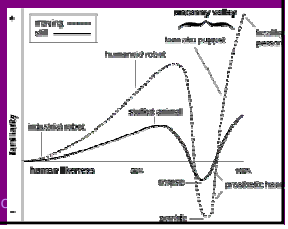
Sensible Assistance

- $S_{Sensible\text{-}Actions}$ Whether the proposed actions make sense
  - See Henry Lieberman's talk on Wednesday
  - Also, Scott Wallace's "similarity" between parties
  - Related to $E_{Sensible}$ for errors
- "Don't Be Stupid"
  - Not keep asking the same thing over and over
- Requires:
  - $S_{User\text{-}models}$ User modeling, so answers are appropriate
  - $S_{Learning}$ Learning, so answers change



Social Presence

- Users may relate better to animated agents
- But "Uncanny Valley"
  - Theory that if agent looks and behaves too much like a person, but not quite, then much worse
  - Increased by movement
  - Linked to zombies and death
    - Ref: Mori, Masahiro (1970). Bukimi no tani *The uncanny valley* (K. F. MacDorman & T. Minato, Trans.). *Energy, 7*(4), 33–35. (Originally in Japanese)
- $R_{Social\text{-}Presence}$



Summative Measures

- Utility
- Trust
- Performance

Utility

- How much Value is what the assistant does?
- How much Work (effort) does it take?
- UTILITY (usefulness) = $\frac{\text{Value}}{\text{Work}}$
- Value = $F(D_{Hand}, V_{Importance}, E_{Avoided})$
  - D_{Hand} How difficult would the task be to do by hand?
    - Partially $F(T_{By\text{-}hand})$
    - Also difficulty in learning how to do it, etc.
  - $V_{Importance}$ How important is it to do the task?
    - Is this the right task to automate?
  - Errors avoided $E_{Avoided}$
- Work (effort)
  - Partially $T_{Assistant}$
  - Maybe include mental workload, etc.

Trust

- What are the factors that go into Trust?
- (Lots of good talks on this topic Tuesday)
- All of the Error metrics
  - Number, cost of errors
  - Ease, likelihood of correcting errors
  - False positives (false alarms) particularly damaging
- Understandability
  - Visibility of what doing
  - Why doing it
- Maybe all the factors?
  - Not just explanations

Others?

- Wayne Iba lists:
  - Competence
  - Attention
  - Anticipation
  - Persistence
  - Deference
  - Integrity
  - Picking appropriate task to automate
- Christopher Miller, et. al. lists other risks
  - Lack of situation & system awareness
  - Increase in user's mode errors
  - Too much trust can also be bad
  - Automation causes increased workload
- Nadine Richard & Seiji Yamada lists "fun factor"
- Are these covered by the factors?

Issue: Converting Do'ers to Managers

- Converting from Direct Manipulation to managing assistants
- But managing is *hard*
  - The most valuable jobs are managers:
    - Terry S Semel, CEO Yahoo, $230.6 mil
    - Barry Diller, CEO IAC, $156.2 mil
    - …
    - Tiger Woods $80.3 mil
    - …
    - Tom Cruise, $31 million

- People have to learn how to effectively use human helpers
- Also, user may know "right" answer only by constructing it
  - Need to "directly manipulate" to investigate the answer
- Don't assume that converting a task to a managerial one will inherently make it easier!

Perceived Costs and Performance

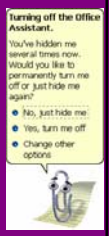
- Alan F. Blackwell, "First steps in programming: A rationale for Attention Investment models. In *Proceedings of the IEEE Symposia on Human-Centric Computing Languages and Environments*, pp. 2-10.
- Given a choice, users evaluate cost-benefit:
  - Investment – learning, etc. to be ready to do task
  - Cost – to do the desired task
  - Pay-off – reduced future cost
  - Risk – probability that no future pay-off will result
  - Decision cost – cost of making this decision
- Users can't know real values, so guess, based on experience, personal style, etc.
  - Easier to estimate the costs to doing task manually
  - Hard to estimate costs and risks of using assistant

Perceived Costs and Benefits

- People overrate errors, under-perceive time saved
  - Strongly prefer not to learn something new
  - Strongly prefer to avoid risk
- People don't necessarily make rational decisions
- User interface can influence perceptions of costs & benefits
  - E.g., Incremental, small steps
- Why there might be a discontinuity in $H_u = f(\ldots)$
  - A little better performance of assistant results in disproportionate gains in H_u

Perceived Costs When Changing

- Particularly difficult with systems that learn
- Past performance may not be a good indicator of future performance
- Need some way to indicate what learned
  - Hopefully more fluid and effective than clippy
  - Continuous instead of binary?

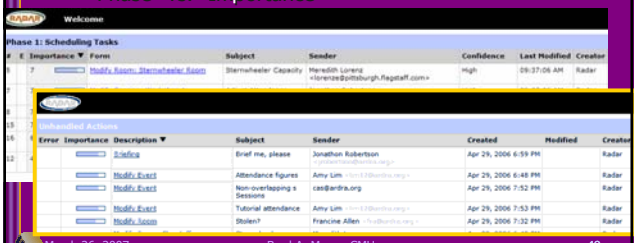


Usability Methods

- Conventional Usability Methods work for Intelligent Assistants
  - Contextual Inquiry
  - Involving designers in the design process
  - Paper-prototyping
  - Wizard-of-Oz prototyping
  - Heuristic analysis
  - Think-aloud user studies
  - Etc.
- Can measure many of the values in A vs. B experiments
  - E.g., compared to the non-assisted version
  - Not appropriate to say "User can easily…" without data

Example

- Improved Radar's task manager through iterative design with user studies
  - Users didn't understand "Confidence", "Phase" vs. "Importance"



Summary: User Happiness?

- H_u = f (….)
- Don't yet know all the factors
- Certainly don't know the function
- But ones that we do know should be measured and optimized
  - Existing HCI methods are effective
- Worthwhile goals to investigate to get assistants that are useful, usable, & pleasant

Acknowledgements

- Thanks to many people who helped with this talk
  - Polo Chau, Andrew Faulring, Michael Freed, Sara Kiesler, Andy Ko, Henry Lieberman, Chris Scaffidi, Dan Siewiorek, Rich Simpson, Stephen Smith, Aaron Steinfeld, Anthony Tomasic, Neil Yorke-Smith, John Zimmerman
- Funded by DARPA through RADAR

User Happiness!

bam@cs.cmu.edu
www.cs.cmu.edu/~bam

$H_u = f(E_{Assistant}\ E_{Negative}\ E_{Positive}\ E_{Value}\ E_{User}\ E_{Corrected}\ E_{By\text{-}hand}\ E_{Cost}\ E_{Avoided}\ E_{Apparentness}\ E_{Correct\text{-}difficulty}\ E_{Sensible}\ W_{Quality}\ W_{Commitment}\ T_{By\text{-}hand}\ T_{By\text{-}Hand\text{-}start\text{-}up}\ T_{By\text{-}Hand\text{-}per\text{-}unit}\ T_{Assistant}\ T_{Training\text{-}start\text{-}up}\ T_{Assistant\text{-}per\text{-}unit}\ T_{Interaction\text{-}per\text{-}unit}\ T_{Monitoring}\ T_{Correcting}\ T_{Responsiveness}\ T_{System\text{-}Training}\ T_{User\text{-}training}\ T_{Average\text{-}for\text{-}each\text{-}correction}\ A_{Error\text{-}rate}\ N_{units}\ P_{Pleasantness}\ U_{Perceive}\ U_{Why}\ U_{Provenance}\ U_{Predictability}\ I_{Assistant\text{-}interfere}\ I_{Screen\text{-}space}\ I_{Cognitive}\ I_{Appropriate\text{-}Time}\ C_{Autonomy}\ C_{Correcting}\ S_{Sensible\text{-}Actions}\ S_{User\text{-}models}\ S_{Learning}\ R_{Social\text{-}Presence}\ D_{Hand}\ V_{Importance})$

Issues Brought Up During Discussion

- Probably calling the top-level measure "Happiness" is incorrect, since users aren't good at perceiving effectiveness
  - What would be a better term for all factors together?
- What about Intelligent Tutors?
  - Need new factors for User's learning, user's motivation
  - The comparison is tutoring by a person
- What about when using Assistant is required, e.g. for safety, by policy?
- What about Mixed Initiative?
  - Are there new factors?
- What are the higher-level, summative factors?
  - $T_{Assistant}$ vs $T_{By\text{-}hand}$; $E_{Assistant}$ vs. E_{User} ; Pleanantness is harder