Language Technology II: Dialogue Design, Evaluation, Learning

Summer 2010

Manfred Pinkal

Outline

- Dialogue Design
- Dialogue Evaluation
- Wizard-of-Oz Experiments
- Dialogue Learning

Best Practise Rules [1]

- Guide the user towards responses that maximize
  - clarity and
  - unambiguousness.
- Allow for the user not knowing
  - the active vocabulary
  - the answer to a question or
  - understanding a question.
- Guide users toward natural 'in vocabulary' responses.
  - *How can I help you? vs.*
  - *Which floor do you want to go?*
  - *You can check an account balance, transfer funds, or pay a bill. What would you like to do?*
- Do not give too many options at once.

Dialogue Design

"...If you'd like to hear all of your options again, press 49. If you've forgotten why you called in the first place, press 50."

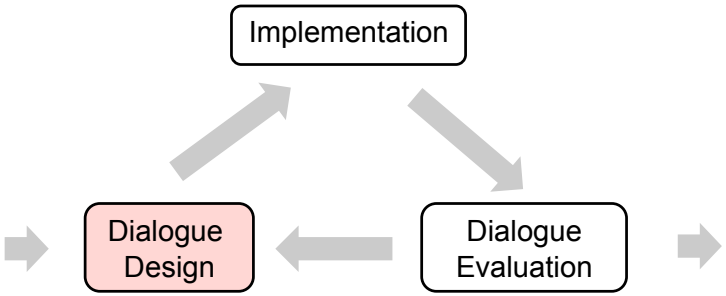
Best Practise Rules [2]

- Keep prompts brief to encourage the user to be brief.
- Supply confirmation messages frequently, especially when the cost or likelihood of a recognition error is high.
- Prefer implicit over explicit grounding.
  - *Is it correct that you need a flight from Frankfurt? vs.*
  - *When do you want to leave from Frankfurt?*
- Use recognizer confidence values to avoid unnecessary grounding steps.

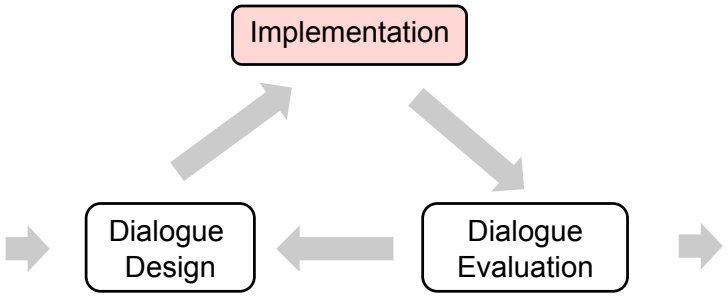
Best Practise Rules [3]

- Assume a frequent user will have a rapid learning curve.
- Allow shortcuts:
  - Switch to expert mode/ command level.
  - Combine different steps in one.
  - Barge-In
- Assume errors are the fault of the recognizer, not the user.
- Allow the user to access (context-sensitive) help at any state.
- Provide escape commands.
- Design graceful recovery when the recognizer makes an error.

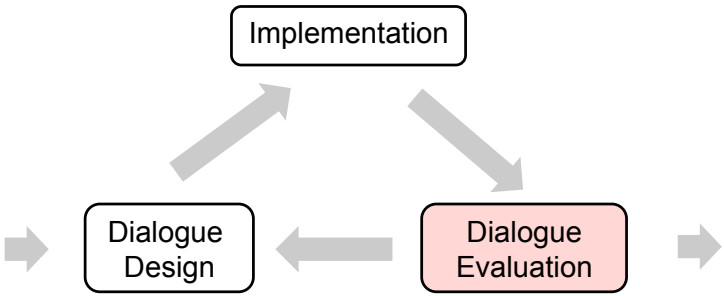
Development Cycle



Development Cycle



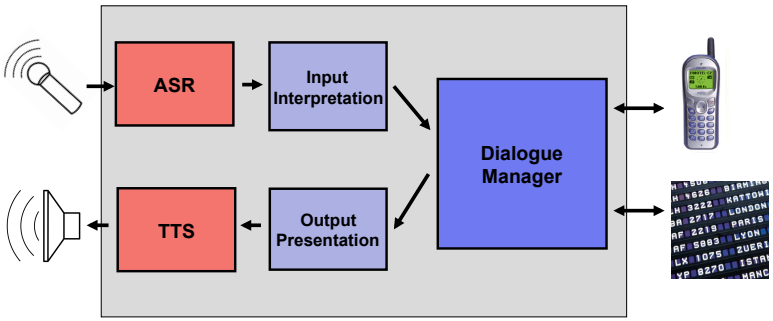
Development Cycle



Levels of Evaluation

- Technical evaluation
- Usability evaluation
- Customer evaluation

Basic Architecture



Levels of Evaluation

- Technical evaluation
  - Typically component evaluation
  - ASR: Word-Error Rate, Concept Error Rate
  - TTS: Intelligibility, Pleasantness, Naturalness
  - Linguistic Coverage: OOV, OOG rate ("out of vocabulary", "out of grammar")
  - Dialogue flow, turn level: Frequency of timeouts, rejects, help requests, barge-ins
- Usability evaluation
- Customer evaluation

Different levels of evaluation

- Technical evaluation
- Usability evaluation
  - Typically an end-to-end “black box” evaluation
  - Main criteria are:
    - Effectiveness (Are dialogue goals accomplished?)
    - Efficiency (Dialogue duration? Number of turns?)
    - User satisfaction
- Customer evaluation

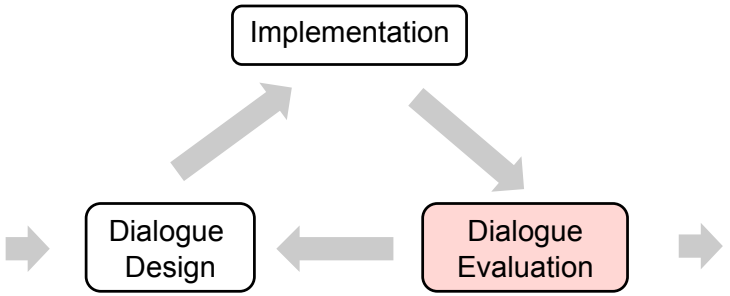
Evaluation of User Satisfaction

- SASSI („Subjective Assessment of Speech System Interfaces“): A Conceptual Framework for designing User Questionnaires
- Dimensions of user satisfaction:
  - System Response Accuracy: User‘s perception of the system as accurate and doing what they expect
  - Likeability: User‘s rating of the system as useful, pleasant, friendly
  - Cognitive demand: The perceived amount of effort needed to interact with the system and feelings arising from this effort
  - Annoyance: User‘s rating of the system as repetitive, boring, irritating, and frustrating
  - Habitability: The extent to which users knew what to do and what the system was doing
  - Speed: How quickly the system responded to user inputs

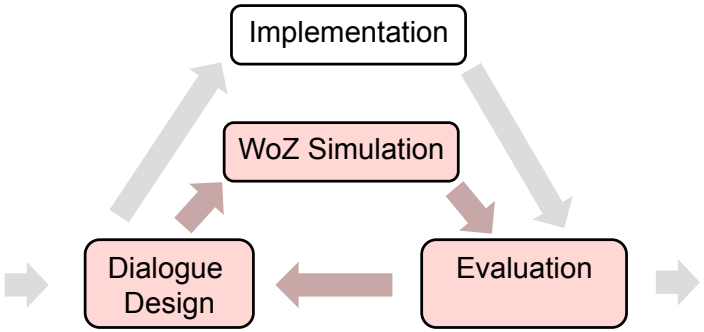
Different levels of evaluation

- Technical evaluation
- Usability evaluation
- Customer evaluation
  - Costs
  - Platform compatibility
  - Maintenance properties
  - Scalability
  - Portability

Development Cycle



Development Cycle



Wizard-of-Oz Studies

- Experimental setup, where a hidden human operator (the “wizard”) simulates (part of) a dialogue system.
- Subjects are told that they interact with a real system.

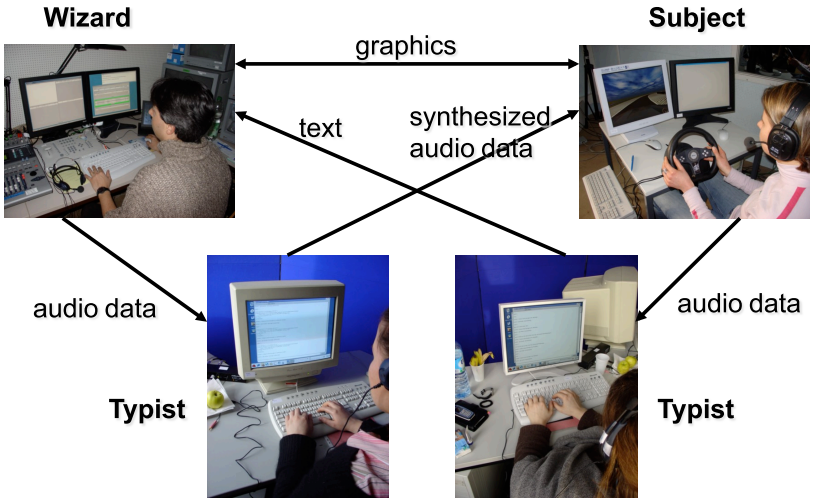
Wizard-of-Oz Studies

- The challenge of providing a convincing WoZ environment:
  - Produce artificial speech output by typing + TTS
  - Induce recognition errors by introducing artificial noise, or presenting input to wizard in a typed version, randomly overwriting single words
  - Constrain natural, conversationally smart wizard reactions by predefining possible system actions and output templates, which the wizard must use.
  - Computer systems are much more efficient in database access, mathematical calculation etc.: Provide the wizard with appropriate interfaces for quick mathematical calculation and database lookup.

An example: WoZ Study in TALK

- Domain: MP3 Player
- Scenario: In-car and In-home
- Multimodal dialogue:
  - Input by speech and ergo-commander/ Keyboard
  - Output by speech and graphics (display)
- Example tasks for subjects:
  - Play a song from the album "New Adventures in Hi-Fi" by REM.
  - Find a song with “believe” in the title and play it.

Information Flow



WoZ Studies: Advantages

- Evaluation of system design at an early stage, avoiding expensive implementation.
- Full control over and systematic variation of speech recognition performance.
- Collection of domain- and scenario-specific language data at an early stage data that can be used
  - for a qualitative analysis of the dialogue behavior of subjects
  - to train or adapt statistical language models
- Systematically exploration of dialogue strategies by varying instructions to the wizard.

Dialogue Policy

- Global design decisions include the specification of
  - a set of dialogue states S, the State Space (a set of nodes in a FSA, or a set of structured information states)
  - a set of possible system actions, the Actions Set (transitions, ISU operations)
  - a range of admissible possible actions A for each state s
- A dialogue policy is a decision procedure that selects specific actions $a \in A$ for possible dialogue states $s \in S$, more technically: a function
  - $\pi: S \rightarrow A$
- Examples for alternatives to be decided by a dialogue policy:
  - Grounding: Explicit grounding act/ implicit grounding act/ no grounding
  - Selection of presentation mode and modality for alternative user options.
- The principal problem of dialogue design:
  - Find a policy π which is optimal with respect to the purposes of the dialogue.

Determining Dialogue Policies

How do we find the optimal dialogue policy?

- Set alternative parameters by hand; examples:

  Minimum confidence value, in dependence of the importance of the decision

  Maximum number of items for which graphical display is appropriate (in dependence of actual user situation)
- Run full implemented system or WoZ experiment with human users, evaluate, modify or refine.

Comment:

- This is the way how things are done in real world, but:
- with high development costs and limited success
- Adaptive dialogue behaviour requires the dynamic combination of a larger number of features.
- Is machine learning and alternative?