Language Technology II: Dialogue Design, Evaluation, Learning

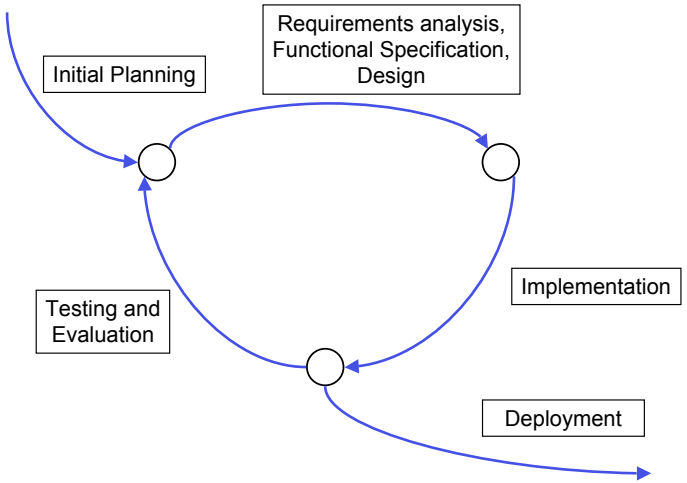
Summer 2008

Manfred Pinkal

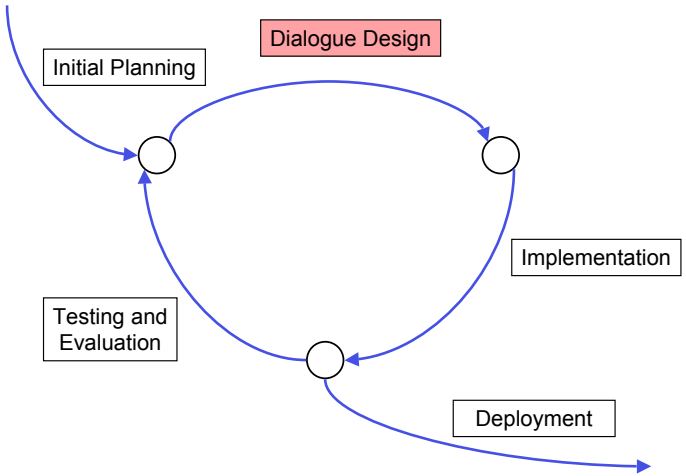
Outline

- Dialogue Design
- Dialogue System Evaluation
  - The Paradise Framework
- Wizard-of-Oz Experiments
- Learning of Dialogue Policies
  - Markov Decision Processes and Reinforcement Learning

Software Development Cycle



Software Development Cycle



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.
- 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.

Best Practise Rules [2]

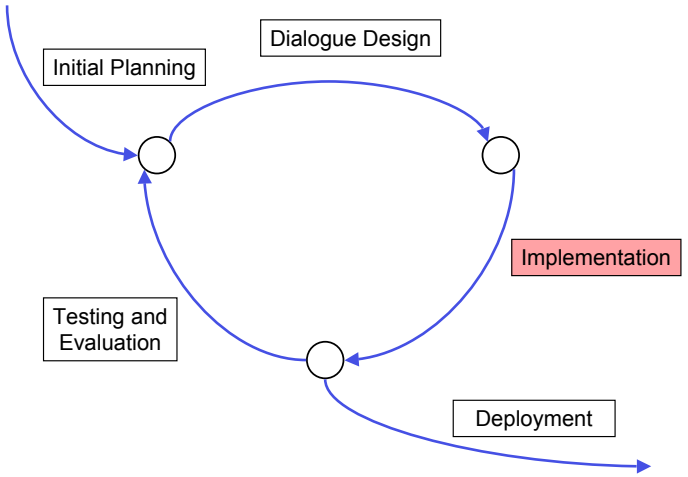
- Prefer implicit over explicit grounding.
  - *So you want to go to the fourth floor? vs.*
  - *I'll take you to the fourth floor.*
  - *Is it correct that you need a flight from Frankfurt? vs.*
  - *When do you want to leave from Frankfurt?*
- Use recognizer confidence values to avoid unnecessary grounding steps.
- 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.

Basic tasks: A Schematic View

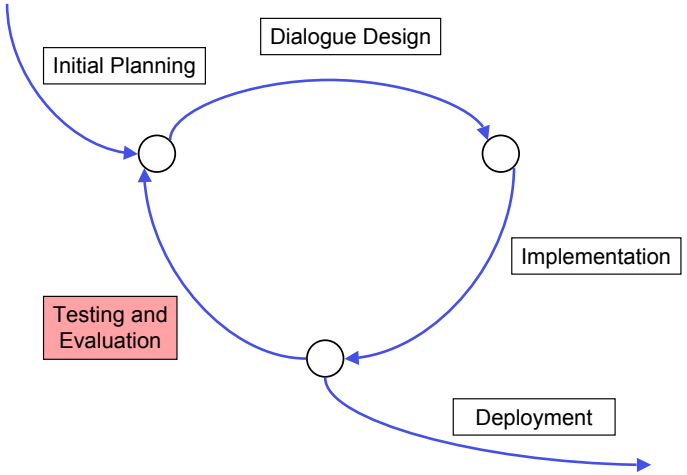
- Global design decisions
  - Specify which information is in principle available to the dialogue system (set of dialogue states S)
  - Specify basic options for system reactions:
  - Range of possible actions A that can be carried out in principle
- Specification of dialogue policy:
  - The action a∈A to be selected given a specific dialogue state s∈S - in other words: a function

    π: S → A

Software Development Cycle



Software Development Cycle



Basic Evaluation Modes

- Technical evaluation
- Usability evaluation
- Customer evaluation

Evaluation Modes

- Technical evaluation
  - Typically component evaluation
  - ASR: Word-Error Rate, Concept Error Rate
  - TTS: Intelligibility, Pleasantness, Naturalness
  - Grammar Coverage, etc.
- Usability evaluation
- Customer evaluation

Different levels of evaluation

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

Different levels of evaluation

- Technical evaluation
- Usability evaluation
- Customer evaluation
  - Leading principle is Return on Investment (ROI)
  - Includes: Costs, Platform compatibility, Maintenance properties

Different levels of evaluation

- Technical evaluation
  - Typically component evaluation
    - ASR: Word-Error Rate
    - TTS: Intelligibility, Pleasantness, Naturalness
    - Grammar Coverage, etc.
- Usability evaluation
  - Typically end-to-end “black box” evaluation
  - Main criteria are:
    - Effectiveness (dialogue goals fully / partially accomplished?)
    - Efficiency (Number of turns? Dialogue duration?)
    - User satisfaction
- Customer evaluation
  - Leading principle is Return on Investment (ROI)
  - Includes: Costs, Platform compatibility, Maintenance properties

Evaluation of User Satisfaction: SASSI

- 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

Usability Evaluation

- Mostly soft criteria:
  - "Usability Guidelines", best-practice rules, form the basis of expert evaluation or user questionnaires.
- Hard, measurable criteria often contradict each other: Systems with high task success may lack efficiency, and vice versa.
- Is it possible to evaluate usability in a objective, predictive, and general way?
- Can we measure user satisfaction?

PARADISE: The Idea

- PARADISE („Paradigm for Dialogue System Evaluation“) is an attempt to provide an objective, quantitative, operational basis for qualitative user assessments
- The foremost criterion for usability evaluation is user satisfaction
  - an intuitive criterion which can not be directly measured, but is only accessible through qualitative user judgments (obtained by user questionnaires, see above).
- User satisfaction is
  - correlated to task success (effectiveness)
  - inversely correlated to the dialogue costs (≈ efficiency)
  - Task success and dialogue costs can be approximated by objective features that can be automatically extracted from logfiles of the dialogue logfiles.

- Reference: M. Walker/ D. Litman/C.Kamm/A.Abella: "PARADISE: A framework for evaluating spoken dialogue agents", Proc. of ACL 1997

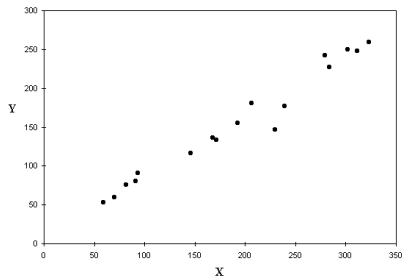
The PARADISE questionnaire

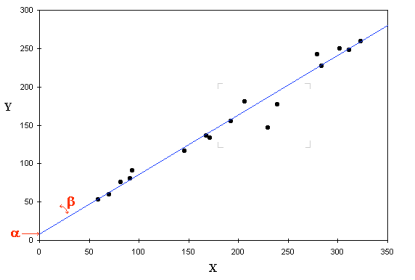
- TTS Performance: Was the system easy to understand?
- ASR Performance: Did the system understand what you said?
- Task Ease: Was it easy to find the information you wanted?
- Interaction Pace: Was the pace of interaction with the system appropriate?
- User Expertise: Did you know what you could say at each point in the dialogue?
- System Response: How often was the system sluggish and slow to reply to you?
- Expected Behaviour: Did the system work the way you expected it to?
- Comparable Interface: How did the system’s vioce interface compare to other systems?
- Future Use: From your current experience with using the system, do you think you would use the system regularly?

PARADISE: Details

- Basis: A set of dialogues (including logfiles) produced by interaction of a dialogue system A with different subjects.
- Asessment of user satisfaction through questionnaire
  - User satisfaction := the arithmetic mean of numeric values assigned to the nine questions of the questionnaire
- Task success information:
  - Either ∈ {0, 1}, Succeed or Fail
  - Or ∈ [0, 1], the proportion of appropriately filled slots (for information-seeking/form-filling dialogue)
  - Or the κ value for agreement between actual and correct slot fillers
- Indicators for dialogue costs:
  - Efficiency measures: Elapsed time, # of System turns, # of user turns
  - Qualitative measures: # of timeout prompts, # of rejects, # of helps, # of cancels, # of barge-ins, mean ASR score
- Compute the best fitting function from task success and dialogue cost information to satisfaction value via linear regression.

Linear Regression





y = ax + b

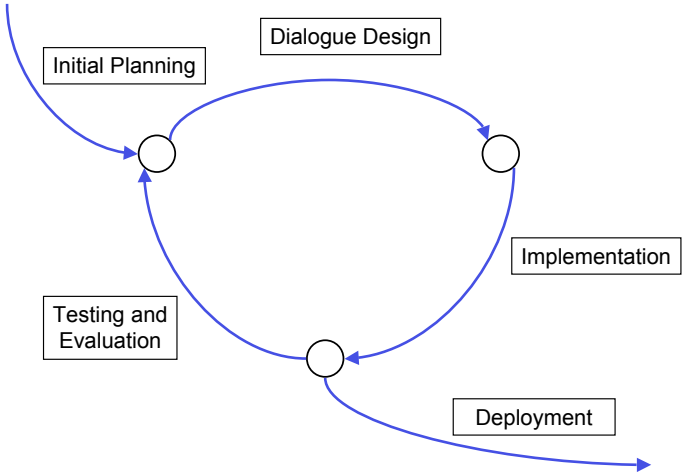
y = 0.5x + 8

The Performance Function

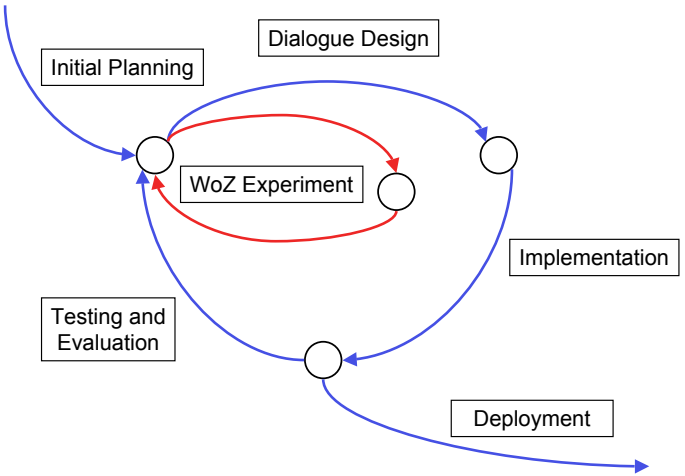
$$US = \left(\alpha * N(\kappa)\right) - \sum_{i=1}^{n} w_i * N(c_i)$$

- *N* is normalisation function based on standard deviation
- $N(\kappa)$ is normalised task success.
- $N(c_i)$ are the normalised cost factors.
- α and w_i are weights on κ and the c_i, determined by linear regression

Software Development Cycle



Wizard-of-Oz Studies



Wizard-of-Oz Studies

- An experimental setup, where a hidden human operator (the “wizard”) simulates (part of) a dialogue system.
- The subjects are left in the belief that they interact with a real system.
- Experimental WoZ systems allow to test a dialogue system (to some extent) before it has been (fully) implemented, thus uncovering basic problems of the dialogue model.
- Also, they are used to collect at an early stage dialogue-specific data that can be used
  - for a qualitative analysis of the dialogue behavior of subjects
  - to train (or manually develop) language models

Wizard-of-Oz Studies

- To adjust WoZ behavior to the behavior of real computer systems is a non-trivial task:
  - Speech output by typing + TTS
  - Human speech recognition outperforms recognition by machines: introduce artificial noise, or present input to wizard in a typed version, randomly overwriting single words
  - Computer systems do not have common-sense or advanced conversational skills: Predefine possible system actions, and output templates for the Wizard
  - Computer systems are much more efficient in database access, mathematical calculation etc.: Provide appropriate interfaces in the WoZ setup

Motivations for WoZ experiments

Standard:

- Evaluate dialogue systems/ dialogue strategies at an early stage, avioding expensive implementation.
- Full control over and systematic variation of speech recognition performance.
- Systematically investigate different dialogue policies by varying instructions to the wizard.

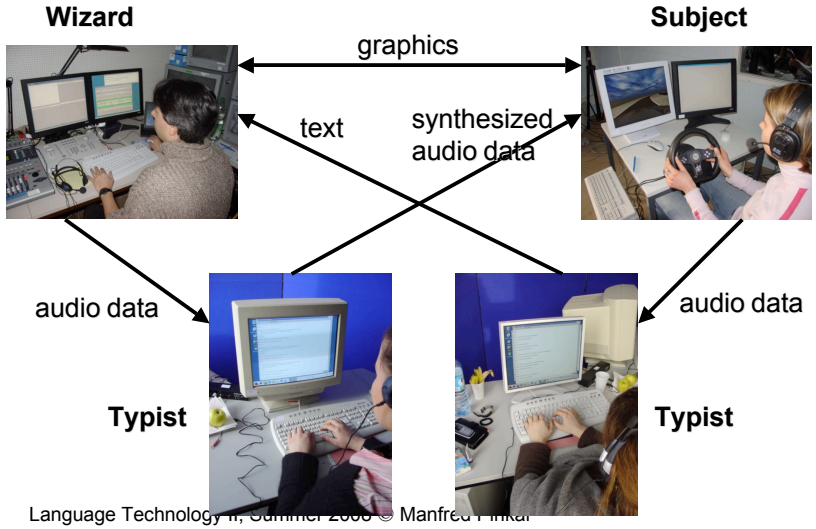
Alternatively:

- Explore a wide variety of dialogue strategies by deliberately vague instructions to the wizard.

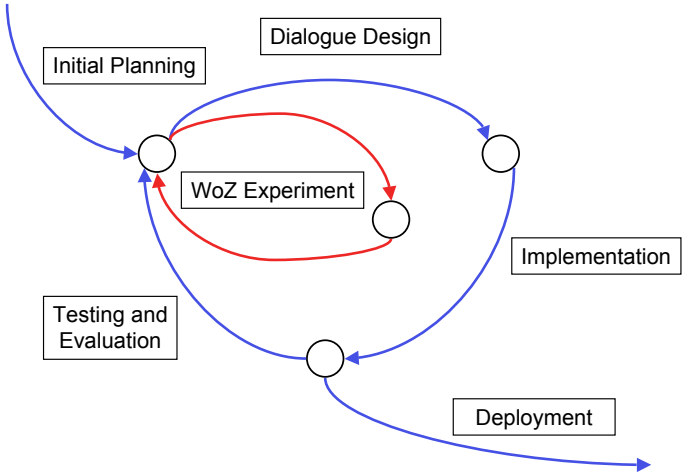
An example: WoZ Study in TALK

- Domain: MP3 Player
- Scenario: In-car and In-home
- Multimodal dialogue:
  - Input by speech and ergo-commander/ Keyboard
  - Output by speech and grahics (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.
- Wizard instruction:
  - Help the user reach their goals

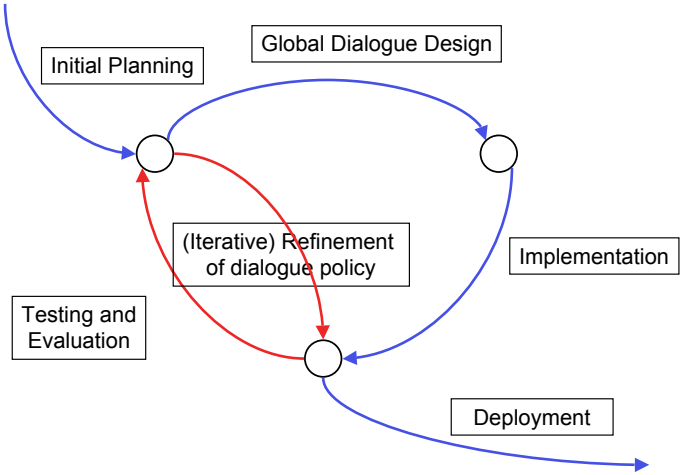
Information Flow in the WoZ Study



Learning Dialogue Strategies



Learning Dialogue Strategies



Dialogue Design

- Global design decisions
  - Specify which information is in principle available to the dialogue system (set of dialogue states S)
  - Specify basic options for system reactions:
  - Range of possible actions A that can be in principle carried out in a state s.
- Underspecified/ non-deterministic dialogue strategy, e.g.:
  - Perform explicit grounding act/ implicit grounding act/ no grounding
  - Ask for further information to reduce number of alternatives/ Display table with alternatives
- Specification of deterministic dialogue policy π: S → A

Determining Dialogue Policies

Option 1:

- Setting thresholds / parameters manually (e.g.: minimum confidence value, maximum number of items to be graphically displayed). Problems:
  - Either: Inadaptive, too little context-sensitivity.
  - Or: A highly complex design task which is difficult to control.

Option 2:

- Supervised learning on WoZ data: Learning average action decisions of wizards for a given state. Problems:
  - Pointwise decisions. - No optimisation on dialogue as a whole.
  - Mimicking of average wizard‘s behaviour: No evaluation of actula behaviour of the wizard, no exploration of new strategies.

Determining Dialogue Policies

Option 3:

- Base local decisions on global performance measures like PARADISE.
  Problems:
  - Measures are available only after the execution of the full dialogue. To evaluate the usefulness or utility of a specific dialogue move, the system must be able to anticipate its impact on the global outcome.
  - The system cannot determine the final outcome: It cannot predict the user's reaction to its action, and thus can only estimate the state resulting from it, which again is input state for its subsequent decision, etc.

  Solution through Reinforcement Learning, which is based on the concept of Markov Decision Process.