Evaluating Dialog Systems

Satanjeev Banerjee
Fall 2003
Papers We Shall See Today

• Empirical Methods for Evaluating Dialog Systems – Paek
  – Talks of the needs and pitfalls of evaluation

  – Presents a formal evaluation framework with some interesting capabilities

• Evaluating Responsiveness in Spoken Dialog Systems – Tsukahara and Ward
  – Presents a novel approach to eliciting utterance by utterance user satisfaction
Empirical Methods for Evaluating Dialog Systems

Tim Paek
Needs for Evaluation

• Does the system meet the goals of the task well?
  – Can I find the # of the bus from CMU to the airport?
  – Has the designer foreseen all my possible needs?

• Compare two systems
  – If possible, across domains!

• Identify system components that can be improved
  – Can we tell that recognition is the problem?
  – Perhaps the repair mechanism is the problem?

• Discover tradeoffs or correlations between factors
  – # utterances correlates well with # repairs
Problems in Evaluation

• Scenario dictates evaluation technique:
  – If user wants to quickly find the next bus to the airport, then metric should be length of dialog
  – If user is elderly, concept error rate should be the metric

• Comparing across systems is tricky:
  – E.g.: how to compare a system that allows barge-in to a system that does not?

• Need a baseline to compare evaluation to.
Baseline: Wizard of Oz
(why this name?)

• Not baseline in the “must beat it” sense 😊
• More like a “gold standard”.
• WoZ study can reveal human’s strategies that the designer might want to model.
How to Conduct a WoZ Study

• Select a metric for evaluation
  – Like length of dialog

• Select the component to be varied
  – Like repair mechanism.

• Hold every other component fixed
  – Like speech recognition. Make wizard use ASR to “hear”, and use different repair mechanisms

• Goal: See how well human does, then compare system’s performance to human’s
Example of Using the WoZ Results

Metric for evaluation

Figure 2. Comparison of two dialog systems with respect to the gold standard.
Example of Using the WoZ Results

Metric for evaluation

Component varied = repair strategy
Example of Using the WoZ Results

Figure 2. Comparison of two dialog systems with respect to the gold standard.
Example of Using the WoZ Results

Metric for evaluation

Switch over point

Component varied = repair strategy

Hard for wizard even!

Figure 2. Comparison of two dialog systems with respect to the gold standard.
Benchmark Complexity

Quantifies the difficulty for a wizard.

$BC = nU - \sum_{x=0}^{n} g(x) = \text{total area of graph} - \text{area under gold curve}$
Precautions for doing WoZ Studies

• System must be modular, or else hard to avoid confounding effects
• Gold is gold because of its quality, not because there is a human in the loop
  – DTMF may be gold standard in task completion (yuck!)
• Must have enough data to make claims!
• Must use multiple wizards.
• Heteroscedasticity, anyone?!
Summary

• Talks of the needs of evaluation
  – And their pitfalls too!
• Extols the virtues of Wizard of Oz studies
  – But reminds us of their pitfalls!
• Shows us what we can learn from WoZ studies
  – How difficult the domain is
  – What the upper bound on performance is
  – How to perhaps model the system
PARADISE: A Framework for Evaluating Spoken Dialog Agents

Marilyn A. Walker, Diane J. Litman, Candace A. Kamm, Alicia Abella
Goals of PARADISE

• Perform task independent evaluation
  – Measure performance independent of task complexity
  – To predict performance in other domains (really?)
• Evaluate dialogs as well as sub-dialogs
• Correlate performance with external criterion
  – Like user satisfaction
  – Also, learn to predict satisfaction from performance
  – E.g.: “lots of repairs correlates with low satisfaction”
Idea behind PARADISE

• Separately evaluate the ends and the means
  – *ends*: how often are the slots correctly filled?
  – *means*: how “good” was the dialog?
    • # utterances, dialog time, repair ratio, etc.

• Compute how much each of these factors affects user satisfaction
  – How important is dialog time? # utterances?
Performance Objectives & Measures

Decision theoretic approach – define objectives, and measures.

- **MAXIMIZE USER SATISFACTION**
  - **MAXIMIZE TASK SUCCESS**
    - KAPPA
  - **MINIMIZE COSTS**
    - EFFICIENCY MEASURES
    - QUALITATIVE MEASURES
      - NUMBER UTTERANCES, DIALOGUE TIME, ETC.
      - AGENT RESPONSE DELAY, INAPPROPRIATE UTTERANCE RATIO, REPAIR RATIO, ETC.
Task Description using Attributes and Values

• Describe a task as a list of all possible attributes and values that can be communicated

• Running example: Simplified Let’s Go
  – Task: Find out what busses run between stops

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Airport, CMU, Downtown, Squirrel Hill</td>
<td>To system</td>
</tr>
<tr>
<td>Destination</td>
<td>Airport, CMU, Downtown, Squirrel Hill</td>
<td>To system</td>
</tr>
<tr>
<td>Busses</td>
<td>28X, 500, 61C</td>
<td>To user</td>
</tr>
</tbody>
</table>
Scenario Description using Attribute-Value Matrices

- **AVM**: List of attribute/value pairs that must be correctly communicated for a dialog to be successful
- Example: Find a bus from CMU to Squirrel Hill → 61C

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>Value #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>CMU</td>
<td>S2</td>
</tr>
<tr>
<td>Destination</td>
<td>Squirrel Hill</td>
<td>D4</td>
</tr>
<tr>
<td>Bus #</td>
<td>61C</td>
<td>B3</td>
</tr>
</tbody>
</table>

- Provides a complete description of the dialog **goal**
Using the AVMs

• Each scenario has a “key” matrix
  – Can’t have multiple key matrices?
• Each experiment creates an “answer” matrix
• Evaluation: How close is the answer matrix to the key matrix?
  – Thus, not measuring dialog quality etc.
**Example Dialog**

**Scenario: Find a bus from CMU to Squirrel Hill**

S: How can I help you?
U: I want a bus to Squirrel Hill.
S: I heard you say Squirrel Hill. From where do you want to go to Squirrel Hill?
U: Carnegie Mellon University
S: I heard you say downtown. Bus # 500 goes from downtown to Squirrel Hill.
S: How can I help you?

“Answer” matrix:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>Value #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Downtown</td>
<td>S3</td>
</tr>
<tr>
<td>Destination</td>
<td>Squirrel Hill</td>
<td>D4</td>
</tr>
<tr>
<td>Bus #</td>
<td>500</td>
<td>B2_22</td>
</tr>
</tbody>
</table>
## Compare Key and Answer

### Key:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>Value #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>CMU</td>
<td>S2</td>
</tr>
<tr>
<td>Destination</td>
<td>Squirrel Hill</td>
<td>D4</td>
</tr>
<tr>
<td>Bus #</td>
<td>61C</td>
<td>B3</td>
</tr>
</tbody>
</table>

### Answer:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
<th>Value #</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Downtown</td>
<td>S3</td>
</tr>
<tr>
<td>Destination</td>
<td>Squirrel Hill</td>
<td>D4</td>
</tr>
<tr>
<td>Bus #</td>
<td>500</td>
<td>B2</td>
</tr>
</tbody>
</table>
Creating Confusion Matrix

<table>
<thead>
<tr>
<th>Answers ↓</th>
<th>Key →</th>
<th>Source</th>
<th>Destination</th>
<th>Bus</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>S1</td>
<td>S2</td>
<td>S3</td>
</tr>
<tr>
<td>Source</td>
<td></td>
<td>S1</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Source</td>
<td></td>
<td>S2</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Source</td>
<td></td>
<td>S3</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Source</td>
<td></td>
<td>S4</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Destination</td>
<td></td>
<td>D1</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Destination</td>
<td></td>
<td>D2</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Destination</td>
<td></td>
<td>D3</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Destination</td>
<td></td>
<td>D4</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Bus</td>
<td></td>
<td>B1</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Bus</td>
<td></td>
<td>B2</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
<tr>
<td>Bus</td>
<td></td>
<td>B3</td>
<td>[ ]</td>
<td>[ ]</td>
</tr>
</tbody>
</table>
# Creating Confusion Matrix

<table>
<thead>
<tr>
<th>Key →</th>
<th>Source</th>
<th>Destination</th>
<th>Bus</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S1</td>
<td>S2</td>
<td>S3</td>
</tr>
<tr>
<td>Answers ↓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Source</td>
<td>S1</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>S2</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>S3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>S4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Destination</td>
<td>D1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>D3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>D4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Bus</td>
<td>B1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>B2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>B3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Totals (sum = 150)</td>
<td>10</td>
<td>7</td>
<td>15</td>
</tr>
</tbody>
</table>
Compute Kappa: Measure of Task Success

\[ \kappa = \frac{P(A) - P(E)}{1 - P(E)} \]

- \( P(A) \) = ratio of answer-key agreement
  = sum of leading diagonal / sum of all numbers
- \( P(E) \) = expected chance agreement
  - Can be computed from attribute value distribution
- \( \kappa = 0 \) when \( P(A) = P(E) \): only chance agreement 😞
- \( \kappa = 1 \) when \( P(A) = 1 \): full agreement 😊
Our Kappa

- $P(A) = \frac{\sum_{i=1}^{n} M(i,i)}{T}$
  
  $= (6+4+10+13+9+9+6+8+7+13+13)/150 = 0.6533$

- $P(E) = \sum_{i=1}^{n} \left( \frac{t_i}{T} \right)^2$
  
  $= (10/150)^2 + (7/150)^2 + (15/150)^2 + (18/150)^2 +$
  
  $(15/150)^2 + (12/150)^2 + (12/150)^2 + (11/150)^2 +$
  
  $(13/150)^2 + (19/150)^2 + (18/150)^2$

  $= 0.0856$

$\kappa = \frac{P(A) - P(E)}{1 - P(E)} = \frac{0.6533 - 0.0856}{1 - 0.0856} = \frac{0.5677}{0.9144} = 0.62$
Measuring Dialog Costs

- Costs like # utterances, repair utterances, etc.
- Represented as function over (sub-)dialogs
- To calculate over sub-dialogs, need dialog structure…
- Hence, PARADISE marks utterances with information goals
- This can be used to break a dialog down into segments, and then apply costs to segments
Tagging Our Example Dialog

S1: How can I help you? [S, D, B]
U1: I want a bus to Squirrel Hill. [D, B]
S2: I heard you say Squirrel Hill. From where do you want to go to Squirrel Hill? [D, S]
U2: Carnegie Mellon University [S]
S3: I heard you say downtown. Bus # 500 goes from downtown to Squirrel Hill. [S,D,B]
U3: How can I help you? [S, D, B]

Convention: Greetings get tagged with all goals
Implicit confirmations get tagged with whatever they are confirming.
Figuring out Segments

- S1 – S3: Figuring out S, D, B
- U1 – S2: Figuring out D
- S2 – U2 – S3: Figuring out S
- S3: Communicating B
Applying Costs to (Sub-) Dialogs

• $\# \text{utt}(S1 – S3) = 5$ utts to complete dialog
• $\# \text{utt}(U1 – S2) = 2$ utts to get D
• $\# \text{utt}(S2 – S3) = 3$ utts to get S
• $\# \text{utt}(S2 – S3) = 1$ utts to give B
• Other kinds of costs possible – like $\#$ repairs
• Can have a variety of costs $c_i$
Finally: Creating a Performance Function

- Function: $\kappa \times c_1 \times c_2 \times \ldots \times c_i \rightarrow \text{real num}$
- Real num should be proportional to external criterion, like user satisfaction
- Goal: To find out which of the factors ($\kappa, c_i$) contribute to user satisfaction
- Performance = $(\alpha \times N(\kappa)) - \sum_{i=1}^{n} w_i \times N(c_i)$
  - $N()$ is a normalizing function
  - $\alpha$ is the weight on $\kappa$
  - $w_i$ is the weight on the costs $c_i$
Learning the Weights

• Get many dialogs by different users using the system.
• For each dialog
  – Elicit user satisfaction value $\Rightarrow$ this is the desired output value to learn
  – Annotate dialog to compute $\kappa$ and the various $c_i$
    $\Rightarrow$ these are input values
• “Solve” for $\alpha$ and $w_i$
• Those components with high weights are important for user satisfaction
Summary

• PARADISE separately evaluates task success and dialog quality
• Can measure partial task success
  – Advantage of using AVMs
• Can focus on sub-dialogs
  – Thus can compare various repair strategies, say
• Performance measure combines lots of costs
  – And can tell which ones affect user satisfaction
• But what about domains that do not lend themselves to slot filling – like Zap?!
Evaluating Responsiveness in Spoken Dialog Systems

Wataru Tsukahara, Nigel Ward
A Different Kind of Dialog System

• Goal of dialog system is to motivate, amuse, entertain, hold attention, etc
  – Necessary for children. For the elderly too?
• How to test these qualities?!  
  – No longer testing task success!
• Define responsiveness as saying exactly the right thing at exactly the right time
• Goal: Evaluate “responsiveness”
Why a New Evaluation Method?

• Kappa not applicable, since not directly interested in task success
• # utterances, # repairs also do not directly measure “responsiveness”
• However, PARADISE can handle any metric, so why not use it?
Memory Game

• Student is asked to recall all 29 stations on a train(?) line
• If she gets stuck, she is given hints
• If she’s correct, she gets positive feedback
• Goal: Study this acknowledgement choice.
• Component varied: “acknowledgment choice”
• Metric: User satisfaction
Their Algorithm for Acknowledgment Choice

• Use context and prosody to infer user’s internal state, like happiness
• “Behave like the user” – if happy, sound happy!
• Slow down when user is in trouble
• Show approval when user on track, supportive when stuck
  – But how to show supportiveness? Word choice?
  – Speech out is copy-pasted speech – probably full sentences?
  – Database of speech is marked with “approval” etc?
Users Insensitive to Choice!

- Users subjected to system with algorithmic ack. choice, and system with random choice
- Users didn’t have any strong preference!
- Reasons mooted (but not “proved”):
  - Conversation too fast for them to notice
  - Acks are very short → hard to notice faults
  - Users forget their momentary reactions
    - In that case, why bother improving?
  - Overall interactions short → perhaps longer interactions would create more lasting impressions.
How to Fix the Problem?

• Ask users to pay more attention
  – But this will likely change their behaviour 😞
• Ask users to think aloud while they interact
  – But this will spoil the fast paced interactivity 😞
• Use third party observers
  – But their opinions differ from first party 😞
• So…?
Do “Re-listening”

• First get users to do the dialog normally
• Soon afterwards, get them to listen to a recording with a transcript at hand
  – And get them to rate every acknowledgement on a 7 point scale
• Motivation: They’ll remember their fleeting impressions!!
Results

• After re-listening, ratings on individual aks. were more consistent with overall ratings
• After re-listening, perception of “kindness” (yasashisa) more strongly correlated with preference of system
• Users provided more detailed comments
Summary

• It is generally not easy to elicit utterance by utterance user satisfaction
  – Even in non fast-paced systems.

• This paper presents a method of doing so,
  – Looks like it works (results consistent with expectations, etc)

• More analysis needed to see how good the method really is?
Papers We Saw Today!

- Empirical Methods for Evaluating Dialog Systems – Paek
  - Talks of the needs and pitfalls of evaluation
  - Presents a formal evaluation framework with some interesting capabilities
- Evaluating Responsiveness in Spoken Dialog Systems – Tsukahara and Ward
  - Presents a novel approach to eliciting utterance by utterance user satisfaction