Tracking the User Experience

Silke M. Witt, Ph.D.
Fluential, Inc.
Common challenges in commercial speech systems

- No history of previous events during an interaction
- Error handling is based on local information
- System performance not known before go-live
- Required Client ability to control caller satisfaction versus automation trade-off
Introducing a new metric with two purposes

1. Use metric to aid the interaction manager decision making
2. Automatically assign a user satisfaction score at interaction end
User experience metric

At each dialog turn a user experience metric is calculated

Past errors, setbacks are weighted by a discounting variable

Different event types:
- NoMatch
- Disconfirmation
- NoInput
- Successful recognition event
- Agent requests

If pre-defined threshold is met, the application switches user treatment
Metric as part of interaction management

Metric tracks experience of user as interaction progresses

A metric represents caller’s state of mind to aid interaction manager decision making

Future work: use to influence other interaction manager decision as to slow down prompt speed, change prompting style and more
Parameter Estimation

Define desired experience in terms of event sequences, both good and bad.

- **Event sequences leading to a treatment change:**
  - metric should be above threshold after 1 disconfirmation, 1 agent request and 1 rejection error.

- **Event sequences NOT leading to a treatment change:**
  - metric should stay below threshold for: 1 timeout, 1 successful turn, 1 rejection and 1 agent request.

Parameters can be estimated via an equation system
Correlation between human ratings and automated scores

<table>
<thead>
<tr>
<th>Cohen’s κ</th>
<th>% Agreement between human and machine</th>
<th>% total agreement (up to a difference of 1) between human and machine</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.733</td>
<td>76.6</td>
<td>93.3</td>
</tr>
</tbody>
</table>
Implementation Example 1

- **Significant reduction in error frequency**
- **Increased caller satisfaction**

**Decreased cost:**
- Each saved transfer saves $0.75, for an application with 6 million calls a year that’s a saving of about $100,000.
Example 2: Impact on automation rates

<table>
<thead>
<tr>
<th>Application</th>
<th>Success rate of initial release</th>
<th>Success rate after parameter adjustment</th>
<th>relative improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>57.4%</td>
<td>63.9%</td>
<td>11.3%</td>
</tr>
<tr>
<td>B</td>
<td>5.7%</td>
<td>8.2%</td>
<td>44.4%</td>
</tr>
<tr>
<td>C</td>
<td>10.1%</td>
<td>13.4%</td>
<td>32.6%</td>
</tr>
</tbody>
</table>

Significant automation improvement by changing parameter values
Example 3: Trade-off between caller satisfaction and automation

- High caller satisfaction = low average score at call end
- High automation = a minimum number of failed calls
Summary

Metric describes experience up to current moment in time.

Parameters are configurable at run-time.

Can be used to automatically score all calls.

Using metric to aid interaction management has been shown to improve the automation in a live system.
Q & A

Thank you!

Contact:
switt@fluentialinc.com