Seeing What You Said: How Wizards Use Voice Search Results

Rebecca Passonneau,* Susan Epstein†, Joshua Gordon* and Tiziana Ligorio‡
*Columbia University
† Hunter College of the City University of New York
‡, † The Graduate Center of the City University of New York
Requests for Books by Title

• you wouldn't by any chance have a book on tape called Armageddon the Battle for Germany

• Recognition issues
  – Statistical LM not sufficiently constrained
  – Grammar-based LM too constrained
  – Long titles increase the likelihood of endpointing within the title
  – Background noise on phone lines degrades recognition performance

• Our approach: complementary to speech engineering
ASR/Voice Search Results with Correct Title

• Single candidate
  
  ASR: *delta cry now*
  DB RET: *don’t cry now*

• List
  
  ASR: *deviant aviator*
  DB RET: *deviant behavior*  queen bess daredevil aviator the aviator

  ASR: *false all heiress*
  DB RET: *rebel heiress*  false pretenses for all their lives false colors false impression
ASR/Voice Search Results without Correct Title

• Single candidate
  ASR: secret honor
  DB RET: secret honor
  TITLE: sacred hunger

• List
  ASR: the language this a come wars
  DB RET: the language of clothes the language of threads
          the language of animals the language of love
          the language instinct
  TITLE: the language of sycamores

  ASR: old and all the earth
  DB RET: holding up the earth to the ends of the earth
          song of the earth
          when calls the heart
  TITLE: hope of earth
Questions

This paper:
• Can human wizards find the correct candidate title if it is in the voice search return?
• Can human wizards tell if the voice search return does not contain the intended title?

Next paper:
• How might dialogue systems do well at the above?

Future work:
• What kinds of useful sub-dialogues can take place to resolve competing candidates?
Outline

• Related work
  – Transaction based SDS
  – WOz with ASR or simulated ASR

• CheckItOut
  – Application domain and baseline architecture
  – New architecture

• Ablated WOz: single turn exchange
  – WOz infrastructure
  – Experimental design

• Results
  – Descriptive statistics
  – Linear regression model of wizard behavior

• Conclusion and Future Work
Related Work

• Transaction-based SDS: database backend
  – System initiative, short utterances <-> short and/or structured fields (Raux et al. 2005)
  – Phonological expansion of terms from ASR (e.g., person names in large (250K) DB of surnames (Georgila et al 2003)
  – Utterance classification + tree of query conditions/specifying detailed query information instead of DM (Komatani et al. 2005)

• WOz dialogues with ASR instead of speech signal
  – ASR (Zollo 1999; Skantze 2003)
  – Simulated ASR (Williams and Young 2004; Reiser et al. 2005)
CheckItOut: Library Transactions

- Andrew Heiskell Braille & Talking Book Library
  - Branch of New York City Public Library
  - Branch of National Library Service
- Book transactions
  - Callers order books/cassettes by telephone
  - Orders sent/returned by U.S.P.O.
- CheckItOut Dialog System
  - Based on Loqui Human-Human Corpus (82 recorded and transcribed calls, aligned with the speech signal)
  - Replica of Heiskell Library book catalogue (N=71,166)
  - Mockup of patron database for 5,028 active patrons

12 July 2009  Seeing what you said  8
CheckItOut Dialogue System

Carnegie Mellon University’s Olympus/RavenClaw Framework
IM: Timing of speech events, turn-taking
ASR: Pocketsphinx .5
NLU: Phoenix Grammar Formalism and Parser
Helios: Confidence Annotation
DM: RavenClaw dialog manager
DR: Domain Reasoner
NLG: Natural Language Generation
TTS: Text-to-speech synthesis

recognizing what is said
Natural Language Understanding

BASELINE METHOD
• Statistical language model(s) constrain the ASR
• Phoenix grammar and parser map the ASR output to (deep) semantic frames, e.g., *book request: Grape Expectations*
• Helios confidence annotator ranks the parses
• RavenClaw decides how to respond to user
  – Domain dependent task tree
  – Domain independent error handling/clarification

FUTURE ARCHITECTURE (Based on this experiment)
• Before parsing, query the backend with ASR
• Pass resulting *shallow* semantic frames to RavenClaw
Ablated Wizard-of-Oz

- Wizard can be shown system internal data, e.g., ASR
- Wizard can take system actions, e.g., backend query
- Wizard can send data to user

Galaxy Hub   Wizard Hub/shifted Galaxy Hub
Voice Search

- Ratcliff/Obersh�elp string matching
  \[ RO = \frac{|\text{Matched characters}|}{|\text{Total characters}|} \]
  where \(|\text{Matched characters}| = \text{recursively find longest common subsequence of 2 or more characters}\)

- Four possible return types
  - Singleton: \( RO \geq 0.85 \)
  - Ambiguous list: 2 to 5 candidates, \( 0.85 > RO \geq 0.55 \)
  - Noisy list: 6 to 10 candidates, \( 0.55 > RO \geq 0.40 \)
  - Empty list: \( RO < 0.40 \)
### Voice Search Examples

<table>
<thead>
<tr>
<th>ASR String</th>
<th>Candidate</th>
<th>R/O score</th>
</tr>
</thead>
<tbody>
<tr>
<td>deviant aviator</td>
<td>deviant behavior</td>
<td>0.84</td>
</tr>
<tr>
<td>the aviator</td>
<td></td>
<td>0.69</td>
</tr>
<tr>
<td>queen bess daredevil aviator</td>
<td></td>
<td>0.56</td>
</tr>
<tr>
<td>false all heiress</td>
<td>rebel heiress</td>
<td>0.67</td>
</tr>
<tr>
<td>for all their lives</td>
<td></td>
<td>0.67</td>
</tr>
<tr>
<td>false impression</td>
<td></td>
<td>0.63</td>
</tr>
<tr>
<td>false pretenses</td>
<td></td>
<td>0.63</td>
</tr>
<tr>
<td>false colors</td>
<td></td>
<td>0.62</td>
</tr>
</tbody>
</table>
Communication

- Wizard and user communicate by means of graphical user interfaces
Title Cycle between Wizard and User

1. Wizard hits “next title” button, caller hears TTS request
2. Caller selects title from list, speaks title
3. Title ASR appears on wizard GUI
4. Wizard selects ASR, inserts in query box, sends query
5. Wizard has four options:
   1. Offer one candidate “with confidence” (Sure)
   2. Offer one candidate “tentatively” (Probably)
   3. Select a set of candidates, ask a clarification question (Question)
   4. Give up (Give up)
6. Caller responses (sent to wizard)
   1. Judge offer as correct or incorrect (5.1 or 5.2)
   2. Assess quality of question (“Can answer”, “Cannot answer”, “Problem””Undecided”)
Experimental Design

User read synopses for each title, ordered the titles by user-determined category

User could end session early

- Participants: 7 students, all 21 pairs (5 native English, 1 bilingual Spanish, 1 Romanian)
- Twenty titles per session
- Two sessions per trial, rotating roles in between
- Five trials per pair of students
- \((21 \times 20 \times 2 \times 5) = 4200\) potential exchanges
- Experiment total: 4,172 exchanges
Recognition Performance in Experiment

- Pocketsphinx .5; WSJ semi-continuous acoustic models
- Statistical language model (CMU/CU SLMTK): 7500 titles
- 7 speakers, ~600 distinct titles each (random subset)
- WER increases with title length, and with sparser data
Descriptive Statistics on Results

- Distribution of backend returns
  - Singleton: 45.42%
  - Ambiguous List: 51.75%
  - Noisy List: 2.83%
  - Empty: None

- Wizards are \(~\text{equally correct:} \) 0.69 (A), 0.67(B), 0.66(C), 0.67(D), 0.69(E), 0.69 (F), 0.70(G)

- Responses if correct title present (N=2986/4172; 71.57%)
  - Confident offer: 68.72%
  - Tentative offer: 26.53%  \[95.25\%\]

- Responses if correct title not present (N=1186/4172; 28.43%)
  - Confident offer: 7.78%
  - Tentative offer: 67.71%
  - Question: 22.32%
  - Give up: 2.20%
Distribution of Responses by Wizard

Best wizard: D (fewer “sure” + “probably”)
Worst wizard: F
Features, Learning and Model

• 26 features to predict wizard’s actions (reduced from 60 by manual curation of cross correlations)
• Linear regression to predict wizard actions, represented numerically (1, 2, 3, 4), with 10-fold cross validation
• Significant predictors:
  – Helios confidence (from system NLU, not seen by wizard)
  – Display type (Singleton, Ambiguous List, Noisy List)
  – Speech rate (faster led to lower accuracy)
  – Successful responses in most recent 3 cycles
• Examples of non-predictive variables:
  – Number of sessions to date
  – Number of times wizard asked for a repeat
Conclusions

1. Results demonstrate utility of voice search for transaction-based spoken dialogue systems
   – Simple string matching method often returns the correct title
   – Incorrect responses could be handled by means of existing error handling methods (Bohus 2005)

2. In general, wizards found the correct candidate title if it was in the voice search return
   – More often tentatively for Ambiguous Lists

3. One wizard (D) had significantly fewer incorrect responses, due to her unique ability to detect when correct title was not present
Discussion and Future Work

• Distribution of wizard actions by caller also varied enormously

• Most questions posed by wizards pertained to the words in the ASR (~60%)

  \textbf{ASR:} charge deaf
  \textbf{RET:} a charmed death
  a changed man

  \textbf{Q:} Did you say charmed or changed?

• Extend to full dialogues
Atypical Queries in CheckItOut: Book titles

- A library application with 71,166 book titles
- Vocabulary size: 54,448 words in book titles
- Varied lengths (max=40; mean=5.4; median=4)

(Titles : Word Length) Distribution
FEATURES
1. Display type (single, list, pseudostring, or empty) returned to the wizard
2. Number of requests from wizard to repeat this title
3. Number of title cycle in current session
4. Number of correctly identified titles in current session
5. Average length of candidates in words
6. Character edit distance between hypothesis and candidates
7. Average rarity of words from hypothesis as measured by word presence in database
8. Average number of exact word matches between hypothesis and candidates
9. Number of occurrences of hypothesis words in database
10. Number of titles in database that contains words in the hypothesis
11. Ratio of number hypothesis words in database to number of titles in database that contains words in the hypothesis
12. Maximum number of contiguous exact word matches between a candidate and the hypothesis
13. Location in hypothesis of maximum number of contiguous word matches
14. Maximum size of gaps between exact word matches
15. Number of candidates
16. Character edit distance between contiguous candidates
17. Score returned by the acoustic model
18. Score returned by Helios (syntactic and semantic confidence)
19. Score returned by the syntactic parse
20. Score returned by the language model
21. Number of frames in the parse
22. Average number of gaps in the parse
23. Speaking rate in frames per word
24. Total number of parses
25. Number of words in the parse
26. Number of words in each slot in the parse