Dialogue Systems

Regina Barzilay

April 14, 2004

Commercial Dialogue System

Case Study: *How May I Help You?* (Gorin et al, 1994 –)

- Goal: support user access to AT&T custom services
- Domain Properties: large vocabulary, speech recognition of variable quality
- Task Requirement: highly accurate response

  Can I reverse the charges on this call?
  (redirected to automatic system)

  How do I call to Jerusalem?
  (redirected to an operator)

Commercial Dialogue System

Case Study: *How May I Help You?* (Gorin et al, 1994 –)

- Goal: support user access to AT&T custom services
- Domain Properties: large vocabulary, speech recognition of variable quality
- Task Requirement: highly accurate response

  Can I reverse the charges on this call?
  (redirected to automatic system)

  How do I call to Jerusalem?
  (redirected to an operator)

Design

- Call-router aims to determine call-type
- Mutli-turn dialogue is used for clarification and utterance disambiguation
- Three dialogue strategies are used:
  - Confirmation
  - Clarification
  - Completion

Dialogue Management

1. The system classifies a type of user's utterance
2. Based on the classification results, the request is either addressed on the spot, or the system continues with a next dialogue move

---

Vocabulary Properties

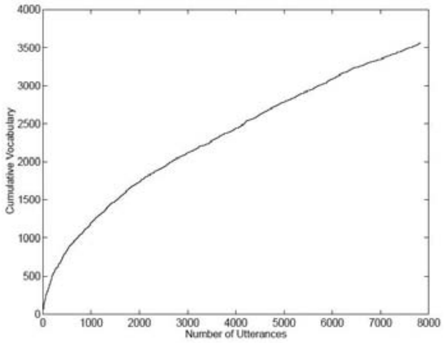


Figure 2. Vocabulary Growth in Database

Example of Confirmation

S: How may I help you?
U: Can you tell me how much it is to Jerusalem?
S: You want to know the cost of the call?
U: Yes, that's right.
S: Please hold on for rate information.

Corpus

- Data: 8K training, 2K testing
- Multi-Label classification (hand labeled): 84% single- and 16% double
- 14 categories and one insertions state

Method

- Supervised classification (ranging from Naive Bayes to AdaBoost)
- Features: ngrams (either of manual transcripts or automatically transcribed)

How Well Does It Work?

- Evaluation Measure: ratio of fully completed dialogs (TASKSUCCESS)
- Evaluation Results: 64% of dialogues are TASKSUCCESS (4774)

Unseen Words

- Out-of-vocabulary-rate — 1.7% , perplexity — 21
- Unseen words include proper names as well as common nouns (*realized*, . . .)

Given a large size of the domain vocabulary, full interpretation of a user utterance is not robust

Results

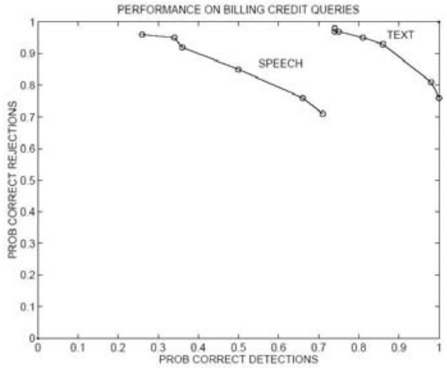


Figure 4. Detecting Billing Credit Queries from Speech

Dealing with Problematic Dialogs

- Automatically identify misunderstandings between a system and the user
- Dynamically change dialogue strategy

Features for Spotting Problematic Dialogs

- **Acoustic/ASR Features**:
  - recog, recog-numwords, ASR-duration, dtmf-flag (a flag for touchstone input), rg-grammar
- **NLU Features**:
  - a confidence measure for all possible tasks that the user could be trying to do
  - salience-coverage (grammar coverage), context-shift, nexttop-task, top-confidence, diff-confidence
- **Dialogue Manager Features**:
  - sys-label, utt-id, prompt, repromt, confirmation, subdial
  - running tallies: num-reprompts, num-confirms, . . .

Typology of Failures

- HANGUP:

S: How may I help you?

U: I need credit please.

S: Okay. What is the reason for the credit?

U: Miss, uh, Miss, different area code I needed.

S: Sorry. I need to know whether you need credit for a wrong number or bad connection.

U: It is a wrong number. (*impatient humming*)

S: What was the number you have dialed?

USER HANGS UP

- WIZARD — task completion requires human intervention
- TASKFAILURE — wrong task is completed

Identification of Problematic Dialogs

- Goal: Learn from data
- Implementation: binary classification
- Classification Features: acoustic, lexical and discourse (manually and automatically extracted)
- Dialogue Length: on average, less than five utterances

Identification of Problematic Dialogues

- Goal: Learn from data
- Implementation: binary classification
- Classification Features: acoustic, lexical and discourse (manually and automatically extracted)
- Dialogue Length: on average, less than five utterances

Examples of Learned Rules

```
if (top-conf <= 0.94)ˆ(dtml-flag = 1)
then problematic;
if (diff-conf <= 0.91)ˆ(asr-duration >= 6.9)
then problematic;
default is tasksuccess
```

Features for Spotting Problematic Dialogs

- **Hand-Labeled Features**:
  - tscript (human transcript), human-label, age, gender, user-modality, clean-tscript (non-word noise removed), cltscript-numwords, rsuccess (map between human and machine label)
- **Whole-Dialogue Features**:
  - num-utts, num-repromts, percent-reprompts, num-confirms, percent-confirms, num-subdials, percent-subdials, dial-duration

Total: 8 features for the whole dialogue and 47 local features

Evaluation Issues

- Prediction vs Identification
- Task-dependent vs task-independent features
- Automatic vs Manual Features

Baseline: 64%

Results

EXCHANGE 1	AUTO	72.3%
	AUTO, TASK-INDEPENDENT	71.6%
	ALL	77%
EXCHANGE 1+2	AUTO	79.9%
	AUTO, TASK-INDEPENDENT	78.6%
	ALL	86.7%
FULL DIALOGUE	AUTO	87%
	AUTO, TASK-INDEPENDENT	86.7%
	ALL	92.3%

Results

- All feature sets improve over the baseline
- Automatic features yield improvement of 23% over the baseline
- Just the first exchange yields 8% improvement over the baseline
- Task-dependent features have a limited contribution in this task (less than 1%)

Dealing with Problematic Dialogs

- Automatically identify misunderstandings between a system and the user
- Dynamically change dialogue strategy

Example

S: Hi, this is AT&T Amtrak Schedule System. This is TOOT. How may I help you?

U: I need to find a train going to New York City.(*ASR OUTPUT: DC I don't care on Thursday.*)

S: Which city do you want to leave from?

U: Baltimore.

S: Do you want me to find the trains from Baltimore to DC on Thursday at anytime now?

U: No.

S: How may I help you?

U: I need to leave on Sunday? (*ASR Output: DC between 1 night and 8*)

Example

*******SECOND ADAPTATION************* S: Which city do you want to go?

U: New York City.

S: Do you want to go to New York City?

U: Yes.

S: Which city do you want to leave from?

U: Baltimore.

S: Do you want to leave from Baltimore?

U: Yes.

S: On which day of the week do you want to leave?

...

Algorithm

```
specify adaptation frequency ‘‘AdaptFreq’’
specify calssification model ‘‘Ruleset’’
specify initial strategy ‘‘CaurStrat’’
for each user utterance
  if ((turns since CurStrat assignment) >= AdaptFreq)
...
CheckRuleset(Ruleset)
 for each rule R in Ruleset
  if (CheckPre(R) == True)
   AdaptStrategy(CurStrat);
  return;
...
AdaptStrategy(CurStrat)
 CurStrat <- MakeConservative (CurStart);
AdaptFreq <- 4;
```

Example

*******FIRST ADAPTATION************* S: I heard you say go to DC between 1 and 8.

U: No. (ASR Output: Help)

S: You can say yes or no.

U: No.

S: How may I help you?

U: I need to go to New York City from Baltimore on Sunday at 8:30 pm. (*ASR Output: To Chicago Tuesday 8:45am*)

S: I heard you say go to Chicago on Tuesday around 8:45 pm. U: No.

Dialogue Strategies for Adaptation

- Initiative:
  - System
  - Mixed
  - User
- Confirmation:
  - Explicit
  - Implicit
  - No

Compare *user initiative with no confirmation* with *system initiative with explicit confirmation*

Evaluation Design

Measure	Adaptive	Non-adaptive
Task Success	0.65	0.23
User Expertise	4	3.2
# of Misrecognized Terms	3.9	6.0
# of System Turns	13.7	17.4

LambdaMoo

Conversations and Grammars

- smalltalk
- personal grammars

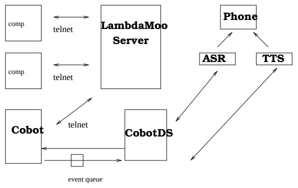
Special Commands

- say
- listen
- summarize

Evaluation Design

- Task: *Try to find a train to New York from Boston at 2:35 pm. If you cannot find an exact match, find the one with the closest departure time. Please write down the exact time of the train you found as well as the total travel time.*
- Measures:
  - Total number of system turns
  - Misrecognized user turns (hand labeled)
  - Success (0, 0.5, 1)
  - User Expertise (1 to 5)
  - User Satisfaction (8 to 40)
- Scope: 4 tasks, 8 users, two version of the system

Exotic Dialogue Systems


comp
telnet
LambdaMoo Server
comp
telnet
Phone
ASR
TTS
Cobot
telnet
CobotDS
event queue