Wiki Bot
Ayushi Aggarwal, Wenxi Lu
Motivation

- Hands-off Wikipedia Search based on Wiki topics

- Multi-lingual search
  - Switch between English and Chinese
What is WikiBot

- Semi-interactive dialogue based search system
- Browser-based
  - Uses browser's built-in Speech Recognition and Text-to-Speech API
- Domain - Wikipedia
Tools

- Built using Javascript
  - Google Speech API
  - Wikipedia API
Architecture

User Speech Input → Google Speech-to-Text → Visual Confirmation → JS → PERFORM ACTION (Search Wiki, Change Volume, Switch language) → Google TTS → Output

Speech to text:
- "Search for dumpling"
- "Volume down"
- "Switch to Chinese"

*https://cloud.google.com/speech/docs/*
Functionality

Current:

● Search Wikipedia
  ○ Search for <insert topic name>
● Switch languages
  ○ Switch to Chinese/已切換到中文
● Volume Control
  ○ Volume up / 增加音量
  ○ Volume down / 减小音量

Extended:

● Search in Chinese
● Switch topics
  ○ Switch topic to <insert topic name>
● Barge-in
  ○ Stop
● Item confirmation
● Item selection
I am a robot who can listen, speak and search for you!

**Status:**

**Speaking...**

**Robot:**

I am a robot who can listen, speak and search for you!

**You:**

**Supported Operations:**

**When in English:**
- Volume up
  - Increase Volume by 4 (0-10 max)
- Volume down
  - Decrease Volume by 4 (0 min)
- Switch to Chinese
  - Switch speech and recognition language to Chinese
- Search for something
  - Search and show a snippet from wikipedia

**When in Chinese:**
- 增加音量
  - 增加4音量 (最大为10) (Volume up)
- 减小音量
  - 减少4音量 (最小为0) (Volume down)
- 使用英文
  - 切换朗读和识别语言为英文 (Switch to English)
Issues and Caveats

- Search is restricted to existing Wikipedia article names
- Search ambiguity - homonyms
- Lack of a barge-in
- Browser-based app - can only be used in Chrome, FireFox, Edge.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Chrome</th>
<th>Edge</th>
<th>Firefox (Gecko)</th>
<th>Internet Explorer</th>
<th>Opera</th>
<th>Safari (WebKit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic support</td>
<td>33[1]</td>
<td>(Yes)</td>
<td>49 (49)[2]</td>
<td>No support</td>
<td>No support</td>
<td>No support</td>
</tr>
</tbody>
</table>
Demo

http://students.washington.edu/wenxil/575/FinalProject_2.html
LING575 - Project Presentation

Analysis of 2000/2001 Communicator Dialogue
Alex Cabral, Nick Chen
Summary

- Overview of data
- Shallow analysis
- Analysis of anger and frustration
Data Overview

- 2000 and 2001 Communicator Dialogue
- Speech only travel planning system
- Simulated
- Nine systems - ATT, BBN, Carnegie Mellon University, IBM, MIT, MITRE, NIST, SRI and University of Colorado at Boulder
- System Improvement between 2000 and 2001
Shallow Analysis

- ASR - similarity between ASR and Transcription
  - Python SequenceMatcher ratio
- System token count
- User token count
- System query repetition (>0.95 SequenceMatcher ratio) against previous two sentences
- Sentiment - Vader
- Average Turns
## Results

For ATT

<table>
<thead>
<tr>
<th></th>
<th>2000</th>
<th>2001</th>
<th>Delta</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interactions</td>
<td>81</td>
<td>158</td>
<td>95%</td>
</tr>
<tr>
<td>ASR</td>
<td>0.86</td>
<td>0.78</td>
<td>-9%</td>
</tr>
<tr>
<td>System Average Length</td>
<td>25.7</td>
<td>23.8</td>
<td>-7%</td>
</tr>
<tr>
<td>User Average Length</td>
<td>2.2</td>
<td>1.4</td>
<td>-36%</td>
</tr>
<tr>
<td>Repetition (last sentence)</td>
<td>31%</td>
<td>63%</td>
<td>103%</td>
</tr>
<tr>
<td>Repetition (last two sentences)</td>
<td>5%</td>
<td>26%</td>
<td>420%</td>
</tr>
<tr>
<td>Sentiment (neg)</td>
<td>0.16</td>
<td>0.14</td>
<td>-13%</td>
</tr>
<tr>
<td>Sentiment (pos)</td>
<td>0.15</td>
<td>0.17</td>
<td>13%</td>
</tr>
<tr>
<td>Sentiment (compound)</td>
<td>0.02</td>
<td>0.01</td>
<td>-50%</td>
</tr>
<tr>
<td>Average Turns</td>
<td>17</td>
<td>33</td>
<td>94%</td>
</tr>
</tbody>
</table>
## Results Summary

### Aggregate

<table>
<thead>
<tr>
<th></th>
<th>MIN</th>
<th>MAX</th>
<th>AVG</th>
<th>MIN</th>
<th>MAX</th>
<th>AVG</th>
<th>Delta Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interactions</td>
<td>59</td>
<td>81</td>
<td>71.88</td>
<td>126</td>
<td>158</td>
<td>143.50</td>
<td>100%</td>
</tr>
<tr>
<td>ASR</td>
<td>0.72</td>
<td>0.89</td>
<td>0.82</td>
<td>0.78</td>
<td>0.94</td>
<td>0.86</td>
<td>5%</td>
</tr>
<tr>
<td>System Average Length</td>
<td>9.1</td>
<td>50.7</td>
<td>25.09</td>
<td>14.1</td>
<td>32.2</td>
<td>21.85</td>
<td>-13%</td>
</tr>
<tr>
<td>User Average Length</td>
<td>2.1</td>
<td>3.9</td>
<td>2.74</td>
<td>1.4</td>
<td>4.2</td>
<td>2.40</td>
<td>-12%</td>
</tr>
<tr>
<td>Repetition (last sentence)</td>
<td>28%</td>
<td>74%</td>
<td>51%</td>
<td>46%</td>
<td>83%</td>
<td>64%</td>
<td>26%</td>
</tr>
<tr>
<td>Repetition (last two sentences)</td>
<td>5%</td>
<td>27%</td>
<td>18%</td>
<td>12%</td>
<td>57%</td>
<td>32%</td>
<td>75%</td>
</tr>
<tr>
<td>Sentiment (neg)</td>
<td>0.04</td>
<td>0.17</td>
<td>0.12</td>
<td>0.04</td>
<td>0.24</td>
<td>0.12</td>
<td>-2%</td>
</tr>
<tr>
<td>Sentiment (pos)</td>
<td>0.15</td>
<td>0.31</td>
<td>0.20</td>
<td>0.17</td>
<td>0.33</td>
<td>0.24</td>
<td>18%</td>
</tr>
<tr>
<td>Sentiment (compound)</td>
<td>0.02</td>
<td>0.10</td>
<td>0.06</td>
<td>0.01</td>
<td>0.13</td>
<td>0.07</td>
<td>8%</td>
</tr>
<tr>
<td>Average Turns</td>
<td>16</td>
<td>20</td>
<td>18.38</td>
<td>20</td>
<td>46</td>
<td>33.50</td>
<td>82%</td>
</tr>
</tbody>
</table>
Analysis of Anger and Frustration

- By conversation and by emotion
- Comparison of anger and frustration to other emotions
- Analysis of both the system and user utterances
- Test the findings and hypotheses of prior work
  - Ang, et. al. (2002) Prosody-Based Automatic Detection of Annoyance and Frustration in Human-Computer Dialog
Analysis of Conversations

- 158 total conversations, 3825 total utterances
- 28 conversations (17.72%), 90 utterances (23.52%) with anger and/or frustration
  - Mean: 3.21
  - Median: 3
  - Max: 8
- 90 angry/frustrated utterances occurred from user having to repeat an utterance
  - 100%
- 15 conversations with 3 or more in a row
  - 16.67%
- 8 conversations with 5 or more in a row
  - 8.89%
Analysis of Emotions

- No difference in length of words or utterances
- “Start over” one of the two most frequent bigrams
- No additional modal verbs
- Very similar results between angry/frustrated and annoyed
  - Annoyed did have more modal verbs
- No initial findings from POS tags
Angry/Frustrated Words
Annoyed Words
Other Emotion Words
Thoughts and Questions

- Findings all seem to be very system-specific
- How viable is it to develop a universal detection methodology?
- Is it important to be able to distinguish annoyed from angry/frustrated?
- Prosodic features seem vital in detecting emotions
Anger Detection

Anna Gale
Overview

- Analysis project looking at detecting anger in the users of a spoken dialog system
- Using the LEGO Spoken Dialogue Corpus (from CMU’s Let’s Go system)
- Looking at prosodic features as well as at least one new discourse-based feature
LEGO Corpus

- Parameterized and annotated version of the CMU Let’s Go database
  - Annotated for emotional state and interaction quality
- Number of Calls: 347
- Number of System-User Exchanges: 9,083
Features

● Prosodic
  ○ Power
  ○ Pitch
  ○ Intensity
  ○ Formants

● Try cosine similarity between current prompt and last two prompts, current response and last two responses
What I set out to do...

- Some sort of **character**-driven, **game**-like application
- All in all, pretty dialog-design-heavy heavy
...and what I ended up doing (not that)

**Multimodal In-Browser Chat System**

- Working name: “flibbertigibbet”
- In essence, a chat room with a *very* simple dialog agent in it
- Type *or* speak to system (and others who are also online)
- System responds to basic social gestures and can tell you the time/date
- It also uses DELPH-IN MRS features to detect how polite you’re being...
Check it out  @  seeve.me
At a glance

**You + Machine**

May 31, 2017

You (spoken) • 1:41

`Hello, what time is it`

Machine • 1:41

Well, hello.

It is currently 36 minutes after 1 o'clock, A.M.

Friend (typed) • 1:43

`B.O.O.I.`

Audio • AutoPlay

Submit Text • Record Speech
Web development-y stuff:

- node.js, socket.io backend

- Standard html, css, javascript/jQuery frontend

- Client-side recording requires a secure https:// connection

- (blah, blah, blah)
Other pieces:

- The node code interfaces with **python** script for getting system’s responses

- Semantic features gotten through ERG API via **pydelphin**

- ...and with **espeak** (for now) for TTS!

- Speech recognition can be either **wit.ai** or **Google Cloud Speech**
Features & issues

More on the python script:

• Replaces interactional models we’ve dealt with
• Gets MRS object (semantic structure) of user’s input
• Detects phrases related to greetings, thanks, farewells (social functions)
• Detects phrases related to asking for the time or date (task functions)
• Only tells you what you want to know if you’re polite enough!
• Responds to all user acts detected
• Browser security measures are a huge pain

• Playing (TTS) sound still doesn’t work on mobile devices

• Interaction still incredibly simplistic

  (I welcome ideas for how to make it less so!)

• But for all the moving pieces (python scripts, espeak TTS, remote ASR services, remote ERG parsing...) it’s surprisingly fast!
Demo...? @ seeve.me
Questions, suggestions?
SPARQL BOT
LING 575 SDS
Will Kearns
RDF

W3C standard for a “smart web” using URIs

Triple store: (subject, predicate, object)

Turtle format (*.ttl):

subject predicate object .

<http://example.org/person/Mark_Twain> <http://example.org/relation/author> <http://example.org/books/Huckleberry_Finn> .
SPARQL

Sparql is a query language for RDF

Example:

prefix reverbDB: <http://server_url/#!>
select ?country ?leader where {
  ?country reverbDB:isacountryin reverbDB:Europe ;
}
reverbDB:Netherlands
Data

Reverb data extraction from wikipedia and the web part of Open IE project

(arg1, relation, arg2)

Converted to RDF and hosted as SPARQL endpoint

<table>
<thead>
<tr>
<th></th>
<th>Reverb</th>
<th>SNOMED CT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tuples</td>
<td>14,728,268</td>
<td>1,360,000</td>
</tr>
<tr>
<td>Entities</td>
<td>2,263,915</td>
<td>327,128</td>
</tr>
<tr>
<td>Predicates</td>
<td>664,746</td>
<td>152</td>
</tr>
</tbody>
</table>

Approach

Query: “What country is in Europe and is east of England”

Decompose: 1) ?w country is in Europe

2) ?w is east of England

Normalize: 1) ?w isacountryin Europe

2) ?w isjusteastof England
Technical Challenges

Alexa Voice Service (AVS) does not provide the user text for a given query (returns intent and slots)

Slot filling in AVS requires manual input

Matching questions pairs against entire database takes $N^2$

Plan to use an inverted index with each query matching at least one term/key term
Limitations & Future Work

Support for federated queries will require linking of resource identifiers, i.e.:

reverbDB:England = dbpedia:England

Many extractions from web have false information, e.g. Obama was born in Kenya

Would like to run OpenIE on trusted sources like Medline Plus or Genetics Home Reference
ProjectDemo-DeniseMak

Meeting start time: Wednesday, May 31, 2017 1:53:05 PM

Organizer: Denise Mak (Steyer Associates Inc)
Table of Contents

● Motivation
● Kitchen Helper
● Code Example
● Features
● Findings
● Demonstration
Motivation

- Referring to culinary resources while cooking is inconvenient
  - Hands may be soiled
  - Hands occupied with other tasks (e.g., cutting, stirring)
  - Last-minute substitutions
  - Last-minute conversions
Kitchen Helper

- Created using Alexa Skills Kit (ASK)
- Accessible via Amazon Echo
- Accessible through a voice user interface (VUI)
- Capabilities
  - Unit conversion
    - Temperature (Celsius/Fahrenheit)
    - Volume/weight (Imperial/metric)
      - Considers a variety of foodstuffs (e.g., flour, sugar)
  - Information lookup
    - Amount of time to cook a given cut of meat
  - Substitution
    - Dry herb to fresh herb
Code Example

1. Initiate Flask app:

```python
app = Flask(__name__)  
ask = Ask(app, "/")
```

2. Create app launch

```python
@ask.launch
def launch():
    hello_msg = render_template('hello')
    reprompt_msg = render_template('hello_reprompt')
    return question(hello_msg).reprompt(reprompt_msg)
```

3. `render_template()` points to yaml file:

Ex:

```
hello: "Kitchen Helper, at your service. What can I do you for?"
hello_reprompt: "You can ask to convert one unit to another, or ask how much juice is in a lemon, lime, or orange."
```
Using Slots

```python
@ask.intent('JuiceIntent', default={"num":"a"})
def juice(fruit, num):
    """Explains how much juice is in a piece of fruit."""
    if fruit == "lemons" or fruit == "lemon":
        factor = 3
    elif fruit == "limes" or fruit == "lime":
        factor = 2
    else:
        factor = 4
    return factor
```
### Matching Intent on Alexa Developer Page

<table>
<thead>
<tr>
<th>Intent Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;I {verb} to know how {quant} {unit} of juice {verb} in {num} {fruit}&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;how {quant} {unit} of juice {verb} in {num} {fruit}&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;how {quant} {unit} of juice {verb} there in {num} {fruit}&quot;</td>
<td></td>
</tr>
<tr>
<td>&quot;please tell me how {quant} {unit} of juice there is in {num} {fruit}&quot;</td>
<td></td>
</tr>
</tbody>
</table>
Features

- Used ngrok for hosting and testing
  - Does not maintain consistent service endpoint URL
    - Need to re-save new URL each time ngrok is run
- Used dummy slots to recognize multiple ways to utter sentence
  - {modal} → will, would
  - {quant} → much, many
- Created functions to convert non-whole numbers from text to speech
  - “Four and a half lbs of pork rib”
  - Several use cases
    - Whole number + fraction
    - Whole number
    - Fraction
Findings (1 of 2)

● Pros
  ○ Easy to use Alexa Skills Kit
    ■ Can specify utterances, confirmations, slots, prompts via graphical user interface (GUI)
  ○ Easy to use Flask
    ■ Built-in functions (e.g., statement, question)
  ○ Easy to test
    ■ Text user interface
    ■ Voice user interface
    ● EchoSim.io
Findings (2 of 2)

- **Cons**
  - Not flexible
    - Sample utterances must be hard-coded
    - Extremely repetitive
    - Testing requires exactitude
  - Difficult to debug
    - GUI does not specify where error occurs while building model
Demonstration
References


Naturalness in Spoken Dialogue: Disfluencies and Backchanneling

Marina Shah
LING 575: Spoken Dialog Systems
May 31, 2017
Motivation

- Adding disfluencies and other ‘content-independent enhancements’ to dialogue creates something that sounds more natural and human
- Two heuristic additions
  - Inserted [well, so, you know, right] at beginnings of sentences sparingly when grammatical
  - Filled pauses & repetitions: No more than 3 per dialogue, max 2 per sentence, heuristically placed where speaker may hesitate (e.g. after “I think”), both can appear together
- Human-rated naturalness
  - With additions, mean naturalness improved by 20%

Figure 4: Mean naturalness across enhancement conditions.

Figure 5: Percentage of participants’ selections of members of the conversation that were correct.
Hypothesis

- Backchanneling and disfluencies are good indicators of human-judged naturalness.
- Disfluencies and similar additions are semantically motivated, so are these enhancements really content-independent?
- Based on Switchboard corpus, frequency of these depends on topic of conversation:
  - Anything from pets to opinions about flag-burning.
  - More controversial/opinionated -> more disfluencies and backchanneling?
Preliminary Stats

● Backchanneling
  ○ 30% of all utterances
  ○ 96.6% happen during longer narratives
  ○ ~3% are turn-passing
  ○ ~.4% are acknowledgment of info
  ○ Most happens after utterances that are 1-15 words long
  ○ More rarely after utterances 60+ words long
  ○ Vast majority follow 1-3 sentence utterances

● Disfluencies
  ○ 9% of all words (almost 8,000 disfluencies)
  ○ 23% are <um, uh, etc.>
  ○ 18% Transition words <you know, well, etc.>
  ○ 27% Conjunctions
  ○ 2% Explanation words
  ○ 28% repeated words
  ○ 1% mumbled/unclear
Proof of Concept

B.72: How is your + what is your feeling about expressing yourself by burning the American flag? / A.73: I'll tell you. / If they didn't give as much coverage to these idiots that burn the flag, it would never happen / do you know what I mean? / B.74: / A.75: It's only because they make a big stink over it. / If they didn't give as much coverage to these idiots that burn the flag, it would never happen. / B.76: / A.77: Yeah. / B.78: you just said it. / A.79: <lipsmack> B.80: It's their right by freedom of speech? / What does speech have to do with / A.81: Yeah. / B.82: burning a flag? / A.83: <breathing> Well it's free. / B.84: <breathing> A.85: that they / I think the idea of freedom of speech goes back to {C and } { [ [ I + [F uh ] } the ] + the ] <breathing> + the ] whole aspect

A.89: ideas {D you know } -/ the + what / the country stands on + America stands on / is that they can do that <breathing> {F Uh, } / though I would never even consider / B.90: <breathing> A.91: in a million years to do it myself / [ [ I + ] + <breathing> I ] think {F uh } {D you / B.92: <breathing> A.93: know } -/ {C but } [ I + ] still / what the [ stan- + flag ] stands for I guess to me is / B.94: <breathing> A.95: that if somebody wants to voice their opinion or display their opinion openly and if that is [ a + a ] way that they can <breathing> show / B.96: <breathing> A.97: their opinion <breathing> then they should be allowed <lipsmack> / {F uh } -/ B.98: {D Now } {F uh } {D Well } [ I + ] still / A.99: [ Unless / B.100: go / A.101: the / B.102: back / A.103: burn + / B.104: to ... /
Proof of Concept

A.17: It's, {uh, } part Chow and part Shepherd / {and } it, / as I understand it, {uh, } both sides [ of the, + ] were thoroughbreds. / {So, } she's a genuine (( Chowperd )). / 
B.18: {Oh, } that sounds interesting. / 
A.19: She has [ the, + the ] color and the black [ to-, + tongue ] of a Chow, / {but, } she has [ the shap-, + the shape ] of the, {uh, } {Shepherd. } / 
B.20: {Oh, } that's, + that's neat. / {How, + about how } big then? / 
A.21: {Oh, } she weighs in at about fifty pounds, / {So } she's a medium size. / 
B.22: Yeah, / yeah. / 
A.23: {But } she's big enough to be intimidating, / 
B.24: Most definitely. / 
A.25: it is a [ fi-, + fixed ] female, by the way, / 
B.26: Yeah. / 
A.27: {and } right from day one, she was teaching me. / 
B.28: {Oh, } I wouldn't doubt it, / yeah. / 
A.29: <Laughter> She's the most intelligent dog I've ever seen. / Course, I'm a little prejudiced, of course. / 
B.30: {Well } that's understandable, / yeah, / it's, {uh, } / 
A.31: <Throat_clearing> {You know, } the first time I brought her home, she was only, {uh, } was it six weeks old. / {And } I spread the newspapers out in the kitchen area. / 
B.32: Uh-huh. / 
A.33: {But, } {uh, } next morning, she let me know in no uncertain terms that she wanted to use the bathroom. / 
B.34: Okay. / 
A.35: {So, } on next night, I spread the newspaper in the bathroom / {and } she used them there. / 
B.36: Oh. / 
A.37: {But } it wasn't too long until she, {uh, } found out she could wait until I let her out in the morning. / 
B.38: Yeah. / 
A.39: {And } since then, / I, + I ] live alone, / 
B.40: Okay. / 
A.41: {and, } {uh, } I live in motor home, / by the way, I'm, {uh, } an [ RV, + full time RV -er, ] / {and } it's, + it's such a pleasure to come home at night / {and } you can see her smiling from ear to ear, she's so happy to see me. / 
B.42: <Laughter> Yeah, / definitely. /
Future Plans

- Backchanneling
  - Compare longer narratives with no backchanneling to more frequent backchanneling
  - Use timed corpus to analyze pause time in between for more information
- Disfluencies & Transitions
  - POS tags and surrounding phrases or parse tree nodes of disfluencies & transitions
- Conversation Topic
  - Perform sentiment analysis
  - Compare strong vs. weak sentiment & good vs. bad
    - Frequency of above phenomena
    - Types of above phenomena
Usefulness

- Create human-like system
- Important to keep types of enhancements consistent with conversation topic
- Examine different regional disfluencies and transition words
References