Speech Processing

Using Speech with Computers
Overview

- **Speech vs Text**
  - Same but different

- **Core Speech Technologies**
  - Speech Recognition
  - Speech Synthesis
  - Dialog Systems
Pronunciation Lexicon

- List of words and their pronunciation
  - (“pencil” n (p eh1 n s ih l))
  - (“table” n (t ey1 b ax l))

- Need the right phoneme set

- Need other information
  - Part of speech
  - Lexical stress
  - Other information (Tone, Lexical accent …)
  - Syllable boundaries
Homograph Representation

- **Must distinguish different pronunciations**
  - ("project" n (p r aa1 jh eh k t))
  - ("project" v (p r ax jh eh1 k t))
  - ("bass" n_music (b ey1 s))
  - ("bass" n_fish (b ae1 s))

- **ASR multiple pronunciations**
  - ("route" n (r uw t))
  - ("route(2)" n (r aw t))
Pronunciation of Unknown Words

- How do you pronounce new words?
- 4% of tokens (in news) are new.
- You can’t synthesis them without pronunciations.
- You can’t recognize them without pronunciations.
- Letter-to-Sounds rules.
- Grapheme-to-Phoneme rules.
Hand written rules

- \([\text{Left Context}] \ X \ [\text{Right Context}] \rightarrow Y\)
- e.g. Pronunciation of letter “c”
  - \(c \ [h \ r] \rightarrow k\)
  - \(c \ [h] \rightarrow ch\)
  - \(c \ [i] \rightarrow s\)
  - \(c \rightarrow k\)
Need an existing lexicon

- **Pronunciations:** words and phones
- **But different number of letters and phones**

Need an alignment

- **Between letters and phones**
- **checked -> ch eh k t**
LTS: alignment

- checked -> ch eh k t

<table>
<thead>
<tr>
<th>c</th>
<th>h</th>
<th>e</th>
<th>c</th>
<th>k</th>
<th>e</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>ch</td>
<td>_</td>
<td>eh</td>
<td>k</td>
<td>_</td>
<td>_</td>
<td>t</td>
</tr>
</tbody>
</table>

- Some letters go to nothing
- Some letters go to two phones
  - box -> b aa k-s
  - table -> t ey b ax-l -
Find alignment automatically

- **Epsilon scattering**
  - Find all possible alignments
  - Estimate $p(L,P)$ on each alignment
  - Find most probable alignment

- **Hand seed**
  - Hand specify allowable pairs
  - Estimate $p(L,P)$ on each possible alignment
  - Find most probable alignment

- **Statistical Machine Translation (IBM model 1)**
  - Estimate $p(L,P)$ on each possible alignment
  - Find most probable alignment
Not everything aligns

- **0, 1, and 2 letter cases**
  - $e \to \varepsilon$ “moved”
  - $x \to k$, $g$-z “box” “example”
  - $e \to y$-uw “askew”

- **Some alignments aren’t sensible**
  - `dept -> d ih p aa r t m ax n t`
  - `cmu -> s iy eh m y uw`
Training LTS models

- Use CART trees
  - One model for each letter

- Predict phone (epsilon, phone, dual phone)
  - From letter 3-context (and POS)

- # # #  c  h  e  c  ->  ch
- # #  c  h  e  c  k  ->  _
- #  c  h  e  c  k  e  ->  eh
- c  h  e  c  k  e  d  ->  k
LTS results

- **Split lexicon into train/test 90%/10%**
  - i.e. every tenth entry is extracted for testing

<table>
<thead>
<tr>
<th>Lexicon</th>
<th>Letter Acc</th>
<th>Word Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>OALD</td>
<td>95.80%</td>
<td>75.56%</td>
</tr>
<tr>
<td>CMUDICT</td>
<td>91.99%</td>
<td>57.80%</td>
</tr>
<tr>
<td>BRULEX</td>
<td>99.00%</td>
<td>93.03%</td>
</tr>
<tr>
<td>DE-CELEX</td>
<td>98.79%</td>
<td>89.38%</td>
</tr>
<tr>
<td>Thai</td>
<td>95.60%</td>
<td>68.76%</td>
</tr>
</tbody>
</table>
For letter V:
if (n.name is v)
  return _
  if (n.name is #)
    if (p.p.name is t)
      return f
    return v
  if (n.name is s)
    if (p.p.p.name is n)
      return f
    return v
return v
But we need more than phones

- **What about lexical stress**
  - \( p \ r \ aa1 \ j \ eh \ k \ t \rightarrow p \ r \ aa \ j \ eh1 \ k \ t \)

- **Two possibilities**
  - A separate prediction model
  - Join model – introduce \( eh/eh1 \) (BETTER)

<table>
<thead>
<tr>
<th></th>
<th>LTP+S</th>
<th>LTPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( L \ no \ S )</td>
<td>96.36%</td>
<td>96.27%</td>
</tr>
<tr>
<td><strong>Letter</strong></td>
<td>---</td>
<td>95.80%</td>
</tr>
<tr>
<td>( W \ no \ S )</td>
<td>76.92%</td>
<td>74.69%</td>
</tr>
<tr>
<td><strong>Word</strong></td>
<td>63.68%</td>
<td>74.56%</td>
</tr>
</tbody>
</table>
Does it really work

- **40K words from Time Magazine**
  - 1775 (4.6%) not in OALD
  - LTS gets 70% correct (test set was 74%)

<table>
<thead>
<tr>
<th></th>
<th>Occurs</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Names</td>
<td>1360</td>
<td>76.6</td>
</tr>
<tr>
<td>Unknown</td>
<td>351</td>
<td>19.8</td>
</tr>
<tr>
<td>US Spelling</td>
<td>57</td>
<td>3.2</td>
</tr>
<tr>
<td>Typos</td>
<td>7</td>
<td>0.4</td>
</tr>
</tbody>
</table>
Spoken Dialog Systems

- **Information giving**
  - *Flights, buses, stocks weather*
  - *Driving directions*
  - *News*

- **Information navigators**
  - *Read your mail*
  - *Search the web*
  - *Answer questions*

- **Provide personalities**
  - *Game characters (NPC), toys, robots, chatbots*

- **Speech-to-speech translation**
  - *Cross-lingual interaction*
Dialog Types

- **System initiative**
  - Form-filling paradigm
  - Can switch language models at each turn
  - Can “know” which is likely to be said

- **Mixed initiative**
  - Users can go where they like
  - System or user can lead the discussion

- **Classifying:**
  - Users can say what they like
  - But really only “N” operations possible
  - E.g. AT&T? “How may I help you?”

- **Non-task oriented**
Let’s Go Bus Information
- 412 268 3526
- Provides bus information for Pittsburgh

Tell Me
- Company getting others to build systems
- Stocks, weather, entertainment
- 1 800 555 8355
SDS Architecture

Recognition → Interpretation → Dialog Manager → Synthesis → Generation
SDS Components

- **Interpretation**
  - Parsing and Information Extraction
  - (Ignore politeness and find the departure stop)

- **Generation**
  - From SQL table output from DB
  - Generate “nice” text to say
Siri-like Assistants

- **Advantages**
  - Hard to type/select things on phone
  - Can use context (location, contacts, calendar)

- **Target common tasks**
  - Calling, sending messages, calendar
  - Fall back on google lookup
“Call John”
“Call John, Bill and Mary and setup a meeting sometime next week about Plan B that’s fits my schedule”
“Make a reservation at a local Chinese restaurant for 4 at 8pm.”
“You should call your mom as its her birthday”
“I have sent flowers to your mom as its her birthday”
Cognitive Assistant that Learns Online

- DARPA project (2003-2008)
- Led by SRI (involved many sites, including CMU)

Personal Assistant that Learns (Pal)

- Answers questions
- Learn from experience
- Take initiative

Spin-off company -> SIRI

- Acquired by Apple in April 2010
SPDA: Platform

- Desktop
  - Computational power

- Phone (non-smartphone)
  - General Magic
    - Was handheld, became phone based
  - Led into GM’s OnStar

- Smartphone
  - Local to device
  - With Cloud
Smartphone + Cloud

- **Smartphone**
  - **Know about user**
    - Contacts, Schedule etc
    - Same speaker
  - **Some computation possible on device**

- **Cloud**
  - **Learn from multiple examples**
  - **Retrain acoustic/language/understanding models**
Voice Search and User Feedback

- **Voice Search**
  - Google, Bing, Vlingo, Apple

- **Get users to help label the data**
  - Listen to user
  - Show best options
    - They select which one is correct

- **Find out how users actually speak**
  - Full sentences vs “search terms”
  - How do English speakers say ethnic names
Voice Search: Simplifications

- **Too many words …**
- **Context**
  - *Where you are (location: home/not home)*
  - *What is on your phone (contacts)*
  - *What you’ve said before*
Personality

- **Have a character**
  - Calls you by name (you choose)
  - Pushy, helpful, nagging ...
  - Allow user choice

- **Personalize it**
  - May form better relationship with it

- **e.g. Siri**
  - US and UK are female/male
Make it do things well

- **Targeted apps**
  - Chose what it will do well

- **Say, 12 different apps**
  - *Have target (hand written) interaction*
  - *Chose what fields you need, and how to interact with the back end data*
  - *If all else fails dump result in Google*

- **Hardware aid**
  - *Infra-red detector for VAD*
Make sure people know it's there

- (Voice search has been on PDA's for years)
- Get a *lot* of people to use it
- Give “silly” examples
  - People will repeat them, you can adapt your system and expect them to say them
Know Your Users

- Young educated
- Standard English speakers
  - (Non-native too?)
- Can you train them to use it better
  - Get them to adapt
Will it work?

- Will people talk in public
  - Talking on the phone is now acceptable
  - Talking to the phone …
- Will people continue to use it
  - Cool at first, but easier to use menus
  - Only use for setting alarms
- Long term use …
- But others may join in anyway
Speech and NLP

- **Same statistical methods**
  - Bayes, n-gram, classification trees

- **NLP in speech**
  - POS tagging (in new languages)
  - Parsing (Syntactic and Prosodic)
  - Information extraction
  - Dialog/Discourse analysis
  - “ASR output” as “noisy” text
Generating Poetry
- Healthcare messages for non-literate
- Appropriate rhyming and cultural references

Emotion ID
- Is this person angry when they are calling us

Singing
Fall Class

Covers

- Speech Recognition, Synthesis, Dialog systems
- Speech ID, evaluation
- Building real systems (ASR, TTS, SDS)
Language Technologies Minor

- 11-721 Grammars and Lexicons
- Plus 3 electives e.g.
  - 11-411 Natural Language Processing
  - 15-492 Speech Processing
  - 11-441 Search Engines and Web Mining
  - Or other LT (Masters) course

- Plus project
  - Often leading to a publication