Multilingual Speech Processing in the context of Under-resourced Languages

Tanja Schultz
Karlsruhe University and Carnegie Mellon University

Tutorial at SLTU ’08, Monday, May 5th 2008, Hanoi
Tutorial Outline

- Introduction and Motivation
  - Motivation
  - History and Leveraged Work
  - Rapid Language Adaptation Server: Spice

- SPICE in detail
  - Text collection & Prompt Selection
  - Phone set specification, Lexical construction
  - ASR Bootstrap & training
  - Language model, TTS Voice building
  - Testing and Tuning

- Latest Experiments and Results
  - Lessons Learnt from past studies

- Conclusions & Next Steps
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?

- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages

- Language Extinction
  - Do we care? What can we do about?

- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts

- Solutions
  - Prior Work: GlobalPhone and FestVox
  - Intelligent Learning Systems
  - Rapid Language Adaptation Server
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?

- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages

- Language Extinction
  - Do we care? What can we do about?

- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts

- Solutions
  - Prior Work: GlobalPhone and FestVox
  - Intelligent Learning Systems
  - Rapid Language Adaptation Server
Many Languages – So What?

Do we really need Speech Processing in many languages?

**Myth:** “Everyone speaks English, why bother?”

- NO: About 6900 different languages in the world
- Increasing number of languages on the web
- Humanitarian and military needs
  - Rural areas, uneducated people, illiteracy

Why is this an research issue?

**Myth:** “It’s just retraining on foreign data – simple!”

- NO: Other languages bring unseen challenges, for example:
  - different scripts, no vowelization, no writing system
  - no word segmentation, rich morphology,
  - tonality, click sounds,
  - social factors: trust, access, exposure, cultural background
Everyone speaks English, why bother?

- Huge number of Languages in the world: 6912
- Language is not only a communication tool but fundamental to cultural identity and empowerment
- Treat linguistic diversity as we treat bio-diversity (David Crystal)
- The strongest eco systems are the most diverse
- Cultures, ideas, memories are transmitted through language

Growing Diversity, no Uniformity!

Diversity of Languages in the Internet grows rapidly
- Top-10: 200%
- All others: 440%
- Chinese: 414%
- Arabic: 940%

### Top Ten Languages Used in the Web

<table>
<thead>
<tr>
<th>Language</th>
<th>% of all Internet Users</th>
<th>Internet Users by Language</th>
<th>Internet Penetration by Language</th>
<th>Internet Growth for Language (2000 - 2007)</th>
<th>2007 Estimate World Population for the Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>31.7%</td>
<td>365,893,996</td>
<td>17.9%</td>
<td>157.7%</td>
<td>042,983,129</td>
</tr>
<tr>
<td>Chinese</td>
<td>31.7%</td>
<td>166,001,513</td>
<td>12.3%</td>
<td>413.9%</td>
<td>1,351,737,925</td>
</tr>
<tr>
<td>Spanish</td>
<td>8.8%</td>
<td>101,539,204</td>
<td>22.9%</td>
<td>311.4%</td>
<td>442,525,601</td>
</tr>
<tr>
<td>Japanese</td>
<td>7.5%</td>
<td>88,300,000</td>
<td>67.1%</td>
<td>83.3%</td>
<td>128,648,245</td>
</tr>
<tr>
<td>German</td>
<td>5.1%</td>
<td>58,981,592</td>
<td>61.1%</td>
<td>112.9%</td>
<td>96,483,326</td>
</tr>
<tr>
<td>French</td>
<td>5.1%</td>
<td>58,456,702</td>
<td>15.1%</td>
<td>379.2%</td>
<td>387,820,873</td>
</tr>
<tr>
<td>Portuguese</td>
<td>4.1%</td>
<td>47,326,760</td>
<td>20.2%</td>
<td>524.7%</td>
<td>234,099,347</td>
</tr>
<tr>
<td>Korean</td>
<td>3.0%</td>
<td>34,120,000</td>
<td>45.8%</td>
<td>79.2%</td>
<td>74,811,268</td>
</tr>
<tr>
<td>Italian</td>
<td>2.7%</td>
<td>31,481,925</td>
<td>52.9%</td>
<td>156.3%</td>
<td>59,546,696</td>
</tr>
<tr>
<td>Arabic</td>
<td>2.5%</td>
<td>28,782,300</td>
<td>8.5%</td>
<td>940.5%</td>
<td>340,548,157</td>
</tr>
<tr>
<td>Rest of World Languages</td>
<td>15.2%</td>
<td>175,474,783</td>
<td>12.4%</td>
<td>440.3%</td>
<td>1,415,478,651</td>
</tr>
<tr>
<td><strong>WORLD TOTAL</strong></td>
<td><strong>100.0%</strong></td>
<td><strong>1,154,358,778</strong></td>
<td><strong>17.8%</strong></td>
<td><strong>219.8%</strong></td>
<td><strong>6,574,668,417</strong></td>
</tr>
</tbody>
</table>

Notes:
1. Internet Top Ten Languages Usage Stats were updated for June 30, 2007.
2. Internet Penetration is the ratio between the sum of internet users speaking a language and the total population estimate that speaks that specific language.
4. World population information comes from the worldpopulationweb site.
5. For definitions and navigation help, see the Site Surfing Guide.
6. Stats may be cited, stating the source and establishing an active link back to Internet World Stats. Copyright © 2007, Minimalls Marketing Group. All rights reserved.
Why Speech Processing?

So we need language support but why *Speech*?

- **Computerization**: Speech is *the* key technology
  - Ubiquitous Information Access: on the go, phone-based
  - Mobile Devices: Too small and cumbersome for keyboards

- **Globalization**:
  - Cross-cultural Human-Human Interaction
    - Multilingual Communities: EU, South Africa, …
    - Humanitarian needs, disaster, health care
    - Military ops, communicate with local people
  - Human-Machine Interfaces
    - People expect speech-driven applications in their mother tongue

⇒ **Speech Processing in multiple Languages**
ML Speech Processing – Research Issue?

It's just retraining on foreign data - no science!

No

New language – new challenges
- Writing system: different or no script, no vowelization, G-2-P
- Word segmentation, morphology
- Sound system: tonals, clicks

No

Different Cultures – social factors
- trust, access, exposure, background

No

Lack of Data and Resources
- Audio recordings, corresponding transcripts
- Pronunciation Dictionaries, Lexicon
- Text corpora, parallel bilingual data

No

Lack of Experts
- Technology experts without language expertise
- Native language experts without technology expertise
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?

- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages

- Language Extinction
  - Do we care? What can we do about?

- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts

- Solutions
  - Prior Work: GlobalPhone and FestVox
  - Intelligent Learning Systems
  - Rapid Language Adaptation Server
Language Characteristics

- **Prosody, Tonality:** Stress, Pitch, Length pattern, Tonal contours
  (e.g. Mandarin 4, Cantonese 8, Thai & Vietnamese 5)

- **Sound system:** simple vs very complex sound systems
  (e.g. Hawaiian 5V+8C vs. German 17V+3D+22C)

- **Phonotactics:** simple syllable structure vs complex consonant clusters
  (e.g. Japanese Mora-syllables vs. German pf, st, ks)

- **Segmentation:** Written form separate words by white space?
  (NO: Chinese, Japanese, Thai, Vietnamese)

- **Morphology:** short units, compounds, agglutination
  English: Natural segmentation into short units – great!
  German: Compounds – not quite so good

  Donau-dampf-schiffahrts-gesellschafts-kapitäns-mütze …

  Turkish: Agglutination – loooooong phrases


  behaving as if you were of those whom we might consider not converting into Ottoman
Writing systems – basic unit is a Grapheme:

Logographic: based on semantic units, grapheme represents meaning
- Chinese: >10,000 hanzi; Japanese ~7000 kanji, Korean to some extend

Phonographic: based on sound units, grapheme represents sound
- Segmental: grapheme roughly corresponds to phonemes
  - Latin (190), Cyrillic (65), Arabic (22) graphems
- Abjads = consonantal segmental phonographic, e.g. Arabic
- Syllabic: grapheme represents entire syllable, e.g. Japanese kana
- Abugidas = mix of segmental and syllabic systems
- Featural: elements smaller than phone, e.g. articulatory features
  - e.g. Korean: ~5600 gulja

Wikipedia: August 2007
Scripts – Examples

How many languages do have a written form?
• Omniglot lists about 780 languages that have scripts
• True number might be closer to 1000

- Logographic scripts, mostly 2 representatives:
  • Chinese: ~ 10,000 hanzi,
  • Japanese: ~7000 kanji (+ 3 other scripts 😊)

- Phonographic:
  • Korean: ~5600 gulja,
  • Arabic, Devanagari, Cyrillic, Roman: ~100 characters
Grapheme-to-Phoneme Relation

Grapheme-to-Phoneme (Letter-to-Sound) Relationship:

- **Logographic**: NO relationship at all
  - concern for Chinese, Japanese, Korean
- **Phonographic**: segmental: close – far – complicated
  - e.g. Finnish, Spanish: more or less 1:1, -- English: try „Phydough“
- **Phonographic**: segmental – consonantal
  - e.g. Arabic: no short vowels written
- **Phonographic**: syllabic
  - e.g. Thai, Devanagari: C-V flips

Automatic Generation of Pronunciations might get complicated
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?
- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages
- Language Extinction
  - Do we care? What can we do about?
- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts
- Solutions
  - Intelligent Learning Systems
  - Prior Work: GlobalPhone and FestVox
  - Rapid Language Adaptation Server
One more Reason for MLSP …

6900 Languages in the world …. BUT

- Extinction of languages on massive scale
  (David Crystal, Spotlight 3/2000)

- Half of all existing languages die out over next century
  ⇒ On Average: Every two weeks one language dies!

- Survey Feb 1999 from Summer Institute of Linguistics

51 languages with 1 speaker left
28 of those in Australia alone
500 languages with < 500 spks
1500 languages with < 1000 spks
3000 languages with < 10,000
5000 languages with < 100,000

96% of world’s languages are spoken by only 4% of its people
The Future of Language

Is a language with 100,000 speakers safe?
- Survival for generations depends on pressure imposed on language
- Dominance of another language, Attitude of the speakers
- Example Breton: beginning of 20th century has 1 Mio speakers, now down to 250,000; Without effort Breton could be gone in 50 years

Reasons that languages die:
- Disaster: Earthquake on Papua New Guinea: Sissano, Warapu, Arop
- Genocide: 90% America’s natives died within 200 years Europeans
- Cultural assimilation: Colonialism, Suppression, Assimilation:
  - (1) Political, social, economic pressure to speak the dominant language,
  - (2) Emerging bilingualism,
  - (3) self-conscious semilingualism, (4) monolingualism

Why should we care?
- Massive death of languages reduces the diversity
- Bio-diversity has been accepted to be a good thing
- Maybe we should accepted this for language diversity (D. Crystal)
What can we do?

What do we learn from other languages?
- Intellectual issues: increase awareness of world history such as movements of early civilization
- Practical issues: medical practices, alternative treatment forms
- Literature … but also new things about the language itself
- Slovakian proverb: “with each newly learned language you acquire a new soul”

How to save endangered languages:
- Community itself must want it, Surrounding culture must respect it
- Funding for courses, materials, and teachers, support the community
- Get linguists into the field, publish information, grammars, dictionaries

Costs associated:
- Depends on conditions (written vs. unwritten languages, etc.)
- Crystal estimates about $80,000 / year per language
- 3000 endangered languages is about $700Mio …
- Organizations to raise funds
  - Foundation of endangered languages (FEL), UNESCO project
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?

- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages

- Language Extinction
  - Do we care? What can we do about?

- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts

- Solutions
  - Intelligent Learning Systems
  - Prior Work: GlobalPhone and FestVox
  - Rapid Language Adaptation Server
Challenges of MLSP

- Lack of Resources: Stochastic approach needs **many** data
  - Hundreds of hours audio recordings and corresponding transcriptions
    Audio data $\leq$ 40 languages; Transcriptions take up to 40x real time
  - Pronunciation dictionaries for large vocabularies (>100,000 words)
    Large vocabulary pronunciation dictionaries $\leq$ 20 languages
  - Mono- and bilingual text corpora: few language pairs, pivot mostly English

- Algorithms are language independent – MLSP is not!
  - Other Languages bring unseen challenges (segmentation, G2P, etc.)
  - Have we already seen ALL or MOST of the language characteristics?

- Social and Cultural Aspects
  - Non-native speech and language, code switching
  - Combinatorical explosion (domain, speaking style, accent, dialect, ...)
  - Few native speakers at hand for minority (endangered) languages
  - Do we have the right data?

- Lack of Language Experts
  - Bridge the gap between technology experts and language experts
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?

- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages

- Language Extinction
  - Do we care? What can we do about?

- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts

- Solutions
  - Intelligent Learning Systems
  - Prior Work: GlobalPhone and FestVox
  - Rapid Language Adaptation Server
⇒ Intelligent systems that learn a language from the user

- Efficient learning algorithms for speech processing
  - **Learning:**
    - Interactive learning with user in the loop
    - Statistical modeling approaches
  - **Efficiency:**
    - Reduce amount of data (save time and costs): at least by factor of 10
    - Speed up development cycles: days rather than months

⇒ Rapid Language Adaptation from universal models

- Bridge the gap between language and technology experts
  - Technology experts do not speak all languages in question
  - Native users are not in control of the technology
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?
- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages
- Language Extinction
  - Do we care? What can we do about?
- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts
- Solutions
  - Intelligent Learning Systems
  - Prior Work: GlobalPhone and FestVox
  - Rapid Language Adaptation Server
Prior Work: GlobalPhone and FestVox
GlobalPhone

Multilingual Database
- Widespread languages
- Native Speakers
- Uniform Data
- Broad Domain
- Large Text Resources
  - Internet, Newspaper

Corpus
- 19 Languages ... counting
- ≥ 1800 native speakers
- ≥ 400 hrs Audio data
- Read Speech
- Filled pauses annotated

Available from ELRA

http://www.cs.cmu.edu/~tanja/GlobalPhone
GlobalPhone Recognizers in 10 Languages

![Error Rate Chart]

- **Word error rate**
- **Phoneme error rate**

<table>
<thead>
<tr>
<th>Language</th>
<th>Word Error Rate</th>
<th>Phoneme Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>JA</td>
<td>10%</td>
<td>33.8%</td>
</tr>
<tr>
<td>DE</td>
<td>11.8%</td>
<td>44.5%</td>
</tr>
<tr>
<td>EN</td>
<td>14%</td>
<td>46.4%</td>
</tr>
<tr>
<td>KO</td>
<td>14.5%</td>
<td>46.1%</td>
</tr>
<tr>
<td>CH</td>
<td>14.5%</td>
<td>45.2%</td>
</tr>
<tr>
<td>TU</td>
<td>16.9%</td>
<td>44.1%</td>
</tr>
<tr>
<td>FR</td>
<td>18%</td>
<td>46.8%</td>
</tr>
<tr>
<td>PO</td>
<td>19%</td>
<td>36.1%</td>
</tr>
<tr>
<td>KR</td>
<td>20%</td>
<td>36.7%</td>
</tr>
<tr>
<td>SP</td>
<td>20%</td>
<td>43.5%</td>
</tr>
</tbody>
</table>
Prior Work: GlobalPhone and FestVox
FestVox: Building Synthetic Voices

⇒ http://festvox.org [Black and Lenzo 2000]

- Documentation, Tools, Scripts, Examples
  - Building Synthetic Voices in the Festival Speech Synthesis System

- Supports:
  - Diphone, unit selection, (later Statistical Parametric Synthesis)
  - Lexicon, letter to sound rules
  - Text processing support.

- Example Languages:
  - CMU development: Croatian, Thai, Chinese (Mandarin), Japanese, Catalan, Spanish, Nepali
  - Non-CMU: Italian, Malay, Maori, Mongolian, Spanish, Telugu, Hindi, Japanese, English (Many), German, Swedish, Polish, …
TTS Build Tasks

Tasks:
- Define phone set, pronunciations (LTS vs. Lexicon)
- Design prompt list, Record data
- Write text front-end (Number, symbol expansion)
- Write/train prosody model
- Deal with peculiarities (segmentation, no vowels, etc.)

Results strongly correlated to effort
- Must-have for funded project
- Involve speech experts
- Almost random distribution rights
  - Others can’t always use the previous results
  - No explicit copyrights (and no way to change them)
- Results often not in format for re-use
CMU projects: Arabic, Thai, Croatian, Farsi
- Shared audio data collection
  - Prompts with phonetic coverage
  - Lots of (ASR) / Single (TTS) speaker(s)
- Shared Phone set
  - Sometimes “similar” e.g. with/without Tone
- Shared Pronunciation Data
  - (Note) input and output are different vocab

*But we need a much tighter coupling ....*
Introduction & Motivation

- Many Languages – so what?
  - Growing Language Diversity on the web
  - Why do we need Speech Processing in many languages?
  - Is this really science – not just retraining on a new language?
- Language Characteristics
  - Written form, scripts, letter-to-sound relationship
  - Issues and Differences between languages
- Language Extinction
  - Do we care? What can we do about?
- Challenges of Multilingual Speech Processing
  - Lack of Resources
  - Lack of Experts
- Solutions
  - Intelligent Learning Systems
  - Prior Work: GlobalPhone and FestVox
  - Rapid Language Adaptation Server
Speech Processing:
Interactive Creation and Evaluation toolkit

- National Science Foundation, Grant 10/2004, 3 years
- Principle Investigator Tanja Schultz, Alan Black

- Bridge the gap between technology experts → language experts
  - Automatic Speech Recognition (ASR),
  - Machine Translation (MT),
  - Text-to-Speech (TTS)

- Develop web-based intelligent systems
  - Interactive Learning with user in the loop
  - Rapid Adaptation of universal models to unseen languages

- SPICE webpage http://cmuspice.org
Speech Processing Systems

Phone set & Speech data

Pronunciation rules

Text data

Input: Speech

AM

Lex

LM

NLP

MT

TTS

Output: Speech & Text

Hello

สวัสดี ด้วย
SPICE Design Principles

1. Data Sharing across Languages
   → Language Universal Models

2. Knowledge Sharing across System Components

Speech-to-Speech Translation

Input $L_s$  Output $L_t$

$AM_s$ $Dict_s$ $LM_s$ $Lex_{st}$ $LM_t$ $Dict_t$ $AM_t$
SPICE: Building Support for a new Language

- SPICE gathers and archives:
  - Appropriate text data
  - Appropriate audio data

- SPICE solicits and defines:
  - Phoneme set
  - Rich prompt set
  - Lexical pronunciations

- ... produces:
  - Pronunciation model
  - ASR acoustic model
  - ASR language model
  - TTS voice

- ... maintains and documents:
  - Projects and users login
  - Data and Models
Tutorial Outline

- Introduction and Motivation
  - Motivation
  - History and Leveraged Work
  - Rapid Language Adaptation Server: Spice

- SPICE in detail
  - Text collection & Prompt Selection
  - Phone set specification, Lexical construction
  - ASR Bootstrap & training
  - Language model, TTS Voice building
  - Testing and Tuning

- Latest Experiments and Results
- Lessons Learnt from past studies
- Future
Welcome to SPICE

Getting started
SPICE is a web-based system for building an end-to-end speech system (including Automatic Speech Recognition and Text-To-Speech) in your own language.

Existing Users
Login with your account:
Login: tanja
Password: **********

New Users
Create a new account:
Login
Password
Re-type Password
Email

Create new account
Login and Project Registration

- Separate "projects" for each language
  - could share info between different projects
- All tasks times are logged
  - Allow us to do cost/efficiency studies
CMU SPICE

SPICE Project

You must do the following to build support for your language:

- Text collection and selection
- Audio collection
- Phoneme set specification
- Lexicon pronunciation creation
- Speech recognition acoustic model creation
- Speech recognition language model creation
- Speech synthesis voice creation
Text collection and selection


Obtain corpus
You can either upload a corpus directly, or crawl the web for one.

- Crawl the Internet. Enter URL: [ ]  Depth: 1 2  [Crawl]
  25% - concatenating
  (view crawl log)
  After clicking "Crawl," the crawl will run on its own in the background. When it has finished, the symbol next to it will turn green.

Note that the text file uploaded must be uncompressed, and 'plain text' (i.e. not a PDF, Word document, etc.)

Obtain prompts
You can upload your own prompts, or use find_prompts to have SPICE automatically generate prompts.

- [ ] [ ]  [Browse...]  [Upload_Prompts]
  Note that the text file uploaded must be uncompressed, and 'plain text' (i.e. not a PDF, Word document, etc.)

- [ ] find_prompts  Find nice prompts  [ ] Check to allow all characters in prompts, leaving text 'as is'

After clicking 'find_prompts,' the prompt finder will run on its own in the background. When it has finished, the symbol next to it will turn green.
Text Collection

- We need text data for the target language
  - Web crawler
  - Plus boost data from similar sites

- Language encoding
  - Non-trivial, but ...
  - Deal with very common alphabets
  - Internally all utf-8

- In-domain vs. general text

- Character analysis
  - Find the character classes:
  - casing, numerals, punctuation etc
Prompt Selection

- Prompts for recording:
  - Collection without transcription
  - “Good” coverage higher chance to give “clean” models

- Prompts should be:
  - Easy to say (no hard words, numerals etc)
    - Contain high frequency words
    - Easy to say in one breath group, 5-15 words
  - “Phonetically” rich / rich in variability
    - But we have no phonetic information yet
    - Make them orthographically rich
    - Greedily select to maximize tri-graphs
If you already have pre-recorded voice data to train Janus Speech Recognition System, and want to create a Janus DB file, please upload it below:

[Upload Audio]

Or, record audio: [Watch Demo Video]

Please read this sentence aloud

Speaker ID: 006
Speaker Name: Tanja Schultz
Prompts Completed: 0 of 4 (NOT-RECORDED)
Speech Data Collection

- Online Audio Recording Tool
  - Collaboratively record large number of speakers
  - Speakers may be separate from developer
- Visual feedback during recording
- Automatic upload on completion
- Java based for portability
  - Works with *many* browsers
- In control of recording
  - We can control the recording format
  - File contents and directory structure
Phoneme set specification

This is a tool which will display all IPA phoneme. As a naive user, you can choose and give names to phonemes you wish your Speech Engine to use. After you have finished, you can click the “Submit” button to create the new acoustic model on the fly.

Consonants (Pulmonic): Please choose the consonant sounds you’d like to have in your new acoustic models by giving it a name in the textbox next to it.

<table>
<thead>
<tr>
<th></th>
<th>Bilabial</th>
<th>Labiodental</th>
<th>Dental</th>
<th>Alveolar</th>
<th>Postalveolar</th>
<th>Retrerval</th>
<th>Palatal</th>
<th>Velar</th>
<th>Uvular</th>
<th>Pharyngeal</th>
<th>Glottal</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>P</td>
<td>p</td>
<td>t</td>
<td>T</td>
<td>t</td>
<td>t</td>
<td>t</td>
<td>k</td>
<td>c</td>
<td>k</td>
<td>q</td>
</tr>
<tr>
<td>f</td>
<td>f</td>
<td>f</td>
<td>d</td>
<td>d</td>
<td>d</td>
<td>d</td>
<td>d</td>
<td>d</td>
<td>g</td>
<td>g</td>
<td>g</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>m</td>
<td>M</td>
<td>m</td>
<td>N</td>
<td>N</td>
<td>Nj</td>
<td>nj</td>
<td>nj</td>
<td>nj</td>
<td>nj</td>
<td>nj</td>
<td>nj</td>
</tr>
<tr>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
<td>r</td>
</tr>
</tbody>
</table>
Phoneme selection

Result: Phoneme set saved in settings
Mapfile saved in KlingonmapFile

Phonemes selected:
P, B, T, D, C, G, M, N, I, U, E, O, A, N
Phoneme Selection

- Selection from standard IPA chart
- User’s names for phonemes
  - Can match user’s lexicon (if one exists)
  - Can match user’s familiarity
- Audio feedback
  - Click to hear recording of each phone
- Allows us to map user’s phone names
  - We map phones to IPA
  - Get phonetic features for user’s phones
  - (what are vowels, what are stops etc)
- Bootstrap from Multilingual Phone Sets
  (see Acoustic Model Initialization)
Lexicon pronunciation creation

Initial Grapheme to Phoneme Rules

Please input an initial Grapheme to Phoneme (G2P) rule of your language.

Based on this rule, our system will "guess" the correct pronunciation of words in your language. You are able to view the predicted pronunciation, change it, delete it, or type a correct pronunciation for this word. The correct pronunciation will be saved into your dictionary and our system will make use of this information to make a better "guess" in predicting pronunciation of new words.

Now please type in Grapheme to Phoneme rule (G2P) for us. Just type one of the most common pronunciation for each grapheme. Thanks.

Upload g2p: [Browse...] [Upload char.info] [Browse...] [Upload]
Pronunciation Dictionary / Lexicon

Input: Speech

Output: Speech & Text

Pronunciation rules

„adios“ → /a/ /d/ /i/ /o/ /s/
„Hallo“ → /h/ /a/ /l/ /o/
现在广播完了 → ???

Hello

Input: Speech

AM Lex LM NLP MT TTS

Swedish ตัวบุคคล

49/80
### Phoneme- vs Grapheme-based ASR

<table>
<thead>
<tr>
<th>Language</th>
<th>Phoneme</th>
<th>Grapheme</th>
<th>Grapheme (FTT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>11.5%</td>
<td>19.2%</td>
<td>18.4%</td>
</tr>
<tr>
<td>Spanish</td>
<td>24.5%</td>
<td>26.8%</td>
<td></td>
</tr>
<tr>
<td>German</td>
<td>15.6%</td>
<td>14.0%</td>
<td>12.7%</td>
</tr>
<tr>
<td>Russian</td>
<td>33.0%</td>
<td>36.4%</td>
<td>32.8%</td>
</tr>
<tr>
<td>Thai</td>
<td>16.0%</td>
<td>18.3%</td>
<td></td>
</tr>
</tbody>
</table>

**Problem:**

- 1 Grapheme ≠ 1 Phoneme

**Flexible Tree Tying (FTT):**

- **One decision tree**
- Improved parameter tying
- Less over specification
- Fewer inconsistencies
* Follow the work of Davel&Barnard

* Word list: extract from text

* G-2-P
  - explicit map rules
  - neural networks
  - decision trees
  - instance learning
    (grapheme context)

* Update after each $w_i$ → effective training
Lexicon pronunciation creation

Rule entry

28.571428571429% Finished

new word:

to

system suggested pronunciation: TO

listen to it

Accept Pronunciation

If you want to skip this word and work on it later, please click Skip this word.

If you don't think it's a valid word in your language, please click Remove this word.

To save your work to resume later and build a lexicon with your current input, click Pause and Build Lexicon.
Issues and Challenges

- How to make best use of the human?
  - Definition of successful completion
  - Which words to present in what order
  - How to be robust against mistakes
  - Feedback that keeps users motivated to continue

- How many words to be solicited?
  - G2P complexity depends on the language (SP easy, EN hard)
  - 80% coverage hundred (SP) to thousands (EN)
  - G2P rule system perplexity

<table>
<thead>
<tr>
<th>Language</th>
<th>Perplexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>50.11</td>
</tr>
<tr>
<td>Dutch</td>
<td>16.80</td>
</tr>
<tr>
<td>German</td>
<td>16.70</td>
</tr>
<tr>
<td>Afrikaans</td>
<td>11.48</td>
</tr>
<tr>
<td>Italian</td>
<td>3.52</td>
</tr>
<tr>
<td>Spanish</td>
<td>1.21</td>
</tr>
</tbody>
</table>
CMU SPICE

Build acoustic model

1. Janus Database Creation
   - Create Janus Database

2. Acoustic Model Initialization
   - Initialization: IN PROGRESS...
   - Initialize_Acoustic_Models

3. Train Context Independent Acoustic Models
   - Computing Labels: IN PROGRESS...
   - Computing Cepstral Means: NOT STARTED YET.
   - Computing LDA matrix: NOT STARTED YET.
   - K-Means Clustering over the codebooks: NOTSTARTED YET.
   - EMI Training: NOT STARTED YET.
   - Train_CL_Models

4. Prepare for Training Context Dependent Acoustic Models
   - Making Polyphone Trees: IN PROGRESS
   - EMI Training over the codebooks: NOT STARTED YET
   - Clustering Contexts: NOT STARTED YET.
   - Performing the Splits: NOT STARTED YET.
   - Prep_CD_Models

5. Train Context Dependent Acoustic Models
   - Context Dependent Acoustic Models Training: IN PROGRESS...
   - Train_CD_Models
Acoustic Model Building

- Acoustic Model Building requires:
  - Transcribed Speech Data
  - Phone set definition
  - Pronunciation Lexicon
    (if transcripts are on word level only – standard case)

- Two step process:
  1. Model Initialization
  2. Model Training
Acoustic Model Initialization

Phone set & Speech data

AM
Lex
LM
NLP
MT
TTS

Input: Speech

Output: Speech & Text
Acoustic Models: Multilingual Data

Step 1:
• Uniform multilingual database (GlobalPhone)
• Build Monolingual acoustic models in many languages
Step 2:
• Combine monolingual acoustic models to a set of multilingual “language independent” acoustic model
Universal Sound Inventory

Speech Production is independent from Language  ⇒  IPA

1) IPA-based Universal Sound Inventory

2) Each sound class is trained by data sharing

- Reduction from 485 to 162 sound classes
- $m,n,s,l$ appear in all 12 languages
- $p,b,t,d,k,g,f$ and $i,u,e,a,o$ in almost all languages
Cross-language Bootstrapping

**Step 3:**
- Define mapping between ML set and new language
- Bootstrap acoustic model of unseen language

Input: Speech

Output: Speech & Text

- **AM**
- **Lex**
- **LM**
- **NLP**
- **MT**
- **TTS**

Example input:
- "hi /h//ai/"
- "you /j/u/"
- "we /w//i/"

Example output:
- "hello"

Example conversion:
- "hi you are I am"

Example output in new language:
- "สวัสดี ครับ"
Build acoustic model

Step 1. Janus Database Creation

- Configure Training

ASR build configured successfully. According to our analysis, you have recorded speech data from 3 speakers amounting to a total of 12.0 minutes.

Total number of iterations to be performed during Step 2 are 5.

1. Iterations 1 - 3: Leave-one-out testing where 2 speakers' data is used for training and 1 speaker's data is used for testing the acoustic models during each round.
2. Iteration 4: 90% of each speaker's utterances are used for training and the rest of the utterances are used for testing the acoustic models.
3. Iteration 5: All the recorded data of the 3 speakers is used for training the acoustic models.
Acoustic Model Building - Configuration

- Checks dependencies and errors
  - Lexicon and phone set correspond
  - Words in recorded prompts are covered by the lexicon

- Divides the recorded data into training and test sets

- Performance evaluation
  - Few data: K-fold cross-validation, with $K = \#\text{speakers}$
  - More data: Data split into 90% (train) and 10% (test)
Step 2. Acoustic Model Training

- Initialization: COMPLETED!
- Computing Labels: COMPLETED!
- Computing Cepstral Means: COMPLETED!
- Computing LDA matrix: COMPLETED!
- K-Means Clustering over the codebooks: COMPLETED!
- EM Training: COMPLETED!
- Segmentation of Speech: COMPLETED!
- Making Polyphone Trees: COMPLETED!
- EM Training over the codebooks: COMPLETED!
- Clustering Contexts: COMPLETED!
- Performing the Splits: COMPLETED!
- Context Dependent Acoustic Models Training: COMPLETED!

View/Hide Results >>
Iteration: 1
# Utterances = 191 Substitutions = 52 Insertions = 64 Deletions = 70 Total Words = 1515 wer = 12.27722772228%

Iteration: 2
# Utterances = 192 Substitutions = 63 Insertions = 160 Deletions = 12 Total Words = 1523 wer = 15.4300722259%

Iteration: 3
# Utterances = 187 Substitutions = 36 Insertions = 14 Deletions = 28 Total Words = 1471 wer = 5.50251529572%

Iteration: 4
# Utterances = 59 Substitutions = 28 Insertions = 19 Deletions = 15 Total Words = 565 wer = 10.9734513274%
Acoustic Model Building - Training

- Requires successful configuration
- Generalized Training Procedure
  - EM Training for Context Independent Models
    - 3-state HMM
    - Number of Gaussians per Model depends on data
  - EM Training for Context Dependent Models
    - Number of models depends on data
  - MFCC front-end, LDA
- Feedback to User:
  - Progress of training procedure
  - Detailed Log files of all processes
  - Results of performance evaluation
Build a language model, click this button:

Build Language Model
Language Model Building

Goal:
- Get as much relevant text data as possible
- Use the text data for
  - Generating recording prompts
  - Generating vocabulary lists
  - Build Language Models for ASR

Approach
1. User supplies an URL to SPICE for crawling
2. Crawler retrieves N documents (web-pages)
3. Compute the statistics (TF-IDF) from the N documents
4. Terms with highest TF-IDF score form query terms
5. Query search engine (Google) to get the URLs for the query terms
6. Crawl the URLs for the data
Building synthesis voice

Tasks:
- recreate [voice (and delete current one)]
- import_waves [waves]
- import_prompts [txt.done.data]
- import_lexicon [lexicon.lexrules]
- label_segments [ ]
- extract_params [no labels]
- build_models [build Spectral and FO models]
- build_dur [build duration model]
- test_voice [test the voice]
- package_voice [package the voice]
Statistical Parametric TTS

- Text-to-speech for Applications, Common technologies:
  - Diphone: too hard to record and label
  - Unit selection: too much to record and label

- Statistical Parametric Synthesis: “just right”
  - “HMM synthesis”: `clustergen` trajectory synthesis
    - Clusters representing context-dependent allophones
    - Works robustly with as little as 10min speech data
    - But … Signal may sound “buzzy”, can lack varied prosody

- Voice Building Process
  - Collect 300-500 utterances from single speaker, rich prompt set
  - Lexical coverage (from Lex Learner)
  - Automatic labeling from acoustic models
  - Automatic: spectral and prosodic models

- More details tomorrow:
  John Kominek: “Synthesizer Voice Quality of New Languages”
Testing ASR-TTS

Simple echo back testing function
SPICE: Demo Tape

SPICE
Speech Processing Interactive
Creation and Evaluation
Toolkit for New Languages

Tanja Schultz, Alan Black
Carnegie Mellon
Tutorial Outline

- Introduction and Motivation
  - Motivation
  - History and Leveraged Work
  - Rapid Language Adaptation Server: Spice

- SPICE in detail
  - Text collection & Prompt Selection
  - Phone set specification, Lexical construction
  - ASR Bootstrap & training
  - Language model, TTS Voice building
  - Testing and Tuning

- Latest Experiments and Results
  - Lessons Learnt from past studies

- Future
Goal: Build Afrikaans – English Speech Translation System with SPICE
- Cooperation with University Stellenbosch and ARMSCOR
- Bilingual PhD visited CMU for 3 month
- Afrikaans: Related to Dutch and English, g-2-p very close, regular grammar, simple morphology

SPICE, all components apply statistical modeling paradigm
- ASR: HMMs, N-gram LM (JRTk-ISL)
- MT: Statistical MT (SMT-ISL)
- TTS: Unit-Selection (Festival)
- Dictionary: G-2-P rules using CART decision trees

Text: 39 hansards; 680k words; 43k bilingual aligned sentence pairs; Audio: 6 hours read speech; 10k utterances, telephone speech (AST)
Good results: ASR 20% WER; MT A-E (E-A) Bleu 34.1 (34.7), Nist 7.6 (7.9)

- Shared pronunciation dictionaries (for ASR+TTS) and LM (for ASR+MT)
- Most time consuming process: data preparation → reduce amount of data!
- Still too much expert knowledge required (e.g. ASR parameter tuning!)

Herman Engelbrecht, Tanja Schultz, Rapid Development of an Afrikaans-English Speech-to-Speech Translator, IWSLT 2005, Pittsburgh, PA, October 2005
SPICE 2007: Field Experiments

- Now targeting more languages in a shorter time frame
- 6-weeks Hands-on Course at CMU in Spring 2007
  - Adopt native languages of participating students as targets
  - Added up to 10 different languages: Bulgarian, English, French, German, Hindi, Konkani, Mandarin, Telugu, Turkish, Vietnamese
- Teams of two students with different native language
- Course goal was to build a simple S-2-S system and use this to communicate with each other in their mother tongue
  - Solely rely on SPICE tools
  - Build speech recognition components in two languages
  - Build simple SMT component in two directions
  - Build speech synthesis components in two languages
- Report back on problems and system shortfalls
The 10 languages cover broad range of peculiarities

**Writing system:**
- Logographic Hanzi (Mandarin);
- Cyrillic (Bulgarian);
- Roman (German, French and English);
- phonographic segmental (Telugu and Hindi);
- phonographic featural (Vietnamese)
- No script: Konkani

**Segmentation:** No segmentation (Chinese); Segmentation white spaces do not necessarily indicate word (Vietnamese)

**Morphology:** simple, low inflecting (English), compounding (German), agglutinating (Turkish) …

**Sound System:** tonal (Mandarin and Vietnamese), stress (Bulgarian)

**G-2-P:** straightforward (Turkish), challenging (Hindi), difficult (English), no relationship (Chinese), invented (Konkani)
SPICE 2007: Lessons Learnt

- It is possible to create speech processing components for 10 languages in 6-weeks using SPICE
- Each language brings new challenges
- Many SPICE features turned out to be very helpful, e.g. only ONE speaker of Konkani in Pittsburgh, web recorder allowed remote collection of more speakers

- Log: time spent in SPICE interface
- Improve interface using breakdown
- Use feedback
- Interface allows for collaborative work

<table>
<thead>
<tr>
<th>Task</th>
<th>Time Spent [hh:mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text Collection</td>
<td>8:35</td>
</tr>
<tr>
<td>Audio Collection</td>
<td>10:07</td>
</tr>
<tr>
<td>Phoneme Selection</td>
<td>4:05</td>
</tr>
<tr>
<td>LM building</td>
<td>1:25</td>
</tr>
<tr>
<td>G-2-P specs</td>
<td>1:30</td>
</tr>
</tbody>
</table>

SPICE 2008: Cross-continental Course

- SPICE-based course between CMU and UKA
  - Students at Carnegie Mellon University, PA
  - Students at Karlsruhe University, Germany
  - Linked by weekly meeting over VC

- Similar to 2007 BUT distributed collaboration
  - Students create ASR & TTS in their native language
  - Bonus for the ambitious: train SMT systems and create a speech-to-speech translation system

- Evaluation includes
  - Time to complete
  - Task difficulties
  - ASR word error rate
  - TTS voice quality

John Kominek, Sameer Badaskar, Tanja Schultz, Alan W Black, IMPROVING SPEECH SYSTEMS BUILT FROM VERY LITTLE DATA, submitted
Tutorial Outline

- Introduction and Motivation
  - Motivation
  - History and Leveraged Work
  - Rapid Language Adaptation Server: Spice

- SPICE in detail
  - Text collection & Prompt Selection
  - Phone set specification, Lexical construction
  - ASR Bootstrap & training
  - Language model, TTS Voice building
  - Testing and Tuning

- Latest Experiments and Results
  - Lessons Learnt from past studies

- Conclusions & Next Steps
Conclusions

- **Challenges in Multilingual Speech Processing**
  - Well defined build processes: ASR, MT, TTS … BUT:
  - Every new language brings unseen challenges
  - Current (statistical) approaches require lots of data
  - … and native language expert and technology expertise
  - How to bridge the gap between language and tech expert?

- **Proposed solution: SPICE**
  - Learning by interaction from a cooperative (but naïve) user
  - Rapid adaptation from language universal models
  - Knowledge sharing across components
  - Development cycle: Days rather than weeks
Next Steps

- **Continuous Server Support**
  - Improve Interface based on user feedback and lessons learned
  - Improve Language Robustness: font encoding, …
  - Software Engineering, Scaling

- **Collaboration**
  - Multiple people working on the same project
  - Leverage from archived projects

- **Cross-confirmation**
  - Multiple views for within and across project confirmation
  - Confidence measure to find appropriate combination

- **Error-blaming**
  - End-to-end system Evaluation vs Component Evaluation
  - Automatic Generation of Recommendations to improve systems
Try This At Home

- System is online at http://cmuspice.org
- Use system for your own project
  - Create new login/passwd and project
- Preloaded Hindi Example
  - Login as
    - Login: demo
    - Passwd: demo
  - Chose project # (your birth day)
- Book on ML Speech Processing