RLAT – Rapid Language Adaptation Toolkit

Tim Schlippe

October 24, 2012
RLAT – Rapid Language Adaptation Toolkit
Outline

• Introduction

• Text Collection and Selection
  – Crawling
  – Snapshot function
  – Recrawling
  – Text normalization
  – Text management
  – Prompt selection

• Language Model Building
  – Language model model management

• Phoneme selection
Outline

• Lexicon Pronunciation Creation
  – G2P Rules
  – Lex Learner
  – Web-derived pronunciations
  – Keynounce – A Game for Pronunciation Generation through Crowdsourcing

• Audio Collection

• Acoustic Model Building
  – Configuration
  – Training

• Testing Automatic Speech Recognition
RLAT – Rapid Language Adaptation Toolkit

- 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

- RLAT webpage: http://csl.ira.uka.de/rlat-dev
• Collects:
  - Appropriate text data
  - Appropriate audio data

• Defines:
  - Phoneme set
  - Rich prompt set
  - Lexical pronunciations

• Produces:
  - Pronunciation model
  - ASR acoustic model
  - ASR language model
  - TTS voice

• Maintains:
  - Projects and users login
  - Data and models
RLAT – Project

--> RLAT project home

**Project Management**

Create new project
To build a new speech system, enter the language the system uses and give your project a name:

- **Language**
- **Project Name**

Create new project

Open existing project
Select an existing project:

- Chinese - 090129_News
- ENGLISH - 080820_NYTimesOlympia
- English - GlobalPhone2_test
- English - SPICE
- French - 081223_Test1
RLAT – Project

--> RLAT project management

Build Your System

- Text and prompt selection (help)
  - Text management
  - SMT-based text normalization (help)
- Audio collection (help)
- Phoneme selection (help)
- Grapheme-to-phoneme rules (help)
- Lexicon pronunciation creation (help)
  - Web-derived pronunciations
- Build acoustic model (help)
- Build language model (help)
  - Language model management
- Test ASR system
- Create speech synthesis voice

User: tschlippe Language: English Project: GlobalPhone2_test [Logout]

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

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

Import data from other projects:

- Text data (copies text folder)
- LM data (copies all lm data)
- Phoneme files (setting, phones, g2p rules)
- Lexicon
- Speech data (creates symbolic link to source project)
- SMT System (import all files)

Source project:

- Chinese - 090129_News
- ENGLISH - 080820_NYTtimesOlympia
- English - GlobalPhone2_test
- English - SPICE
- French - 081223_Test1
Overview – Automatic Speech Recognition

- Front End (Preprocessing)
- Decoder (Search)
- Text
- Acoustic Model
- Lexicon / Dictionary
- Language Model
Overview – Automatic Speech Recognition
RLAT – Text Collection and Selection

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

• Possibilities to obtain text data:
  – Web crawler (web pages and search engines)
  – Upload (local) / download texts (remote)

• Language encoding
  – To deal with very common alphabets internally all utf-8
RLAT – Text Collection and Selection – Crawling

- Interface for automatic text collection
  - Upload function, Web crawler, Recrawl: boost data from similar sites
- Filtering:
  - Language Identification
  - Remove HTML tags, XML, JavaScripts, ...
  - Convert from different character encodings (ISO-8859, UTF8, ...) and file formats into raw text in UTF-8
  - Text Normalization
RLAT – Text Collection and Selection – Crawling

- Link Level Depth, Upload Test File to evaluate text being crawled, Upload File with URLs and Search Terms, Clock ...

```
Obtain corpus
You can either crawl the internet for a corpus ...

- Enter URL: [ ] Remain in web-domain [ ]
- Link Depth: [ ]
- Interval (snapshot period): 24 hours [ ]
- Check if you want RLAT to build LMs automatically [ ]
- Crawl (view crawl log) (view wget log)

- Upload text file with URLs (Note: Each URL in one line):
  - [ ] Durchsuchen [ ] Upload_URLs

- Upload text file with search terms (Note: Each search term in one line):
  - [ ] Durchsuchen [ ] Upload_Search_Terms

- Upload text file from specific domain for informative and visual feedback about the quality of text being crawled (snapshot functionality)
  - [ ] Durchsuchen [ ] Upload_Test_Data

- Upload vocabulary file to determine the case of words from this file rather than statistically (default method). File should contain only one word per line.
  - [ ] Durchsuchen [ ] Upload_Vocabulary_File

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.

13 days 11 hours 48 minutes 35 seconds```
RLAT – Text Collection and Selection – Snapshot function

- **Snapshot functionality**
  - Informative and visual feedback about the quality of text being crawled
  - Results which indicate the quality (PPL, OOV) of the collected texts are computed and displayed periodically (to be defined by the user) during the crawling process

Obtain corpus
You can either crawl the internet for a corpus ...

- Enter URL: www.leasions.de
- Link Depth: 1 2 5 10 20 50
- Interval (snapshot period): No Snapshot 12 Hours 24 Hours 3 Days 7 Days Crawl (view wget log)

Upload text file with URLs (Note: Each URL in one line):
- Link Depth: 1 2 5 10 20
- Interval (snapshot period): No Snapshot 12 Hours 24 Hours 3 Days 7 Days MultiCrawl (view crawl log)

Upload text file with search terms (Note: Each search term in one line):

Upload text file from specific domain for informative and visual feedback about the quality of text being crawled (snapshot functionality)

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.

13 days 11 hours 48 minutes 35 seconds
Language Model Performance of 5-fold cross validation
Generated on: December 02 2009 19:00:04
Perplexity (Average): 54.0476
Perplexity (max): 35.5676
Perplexity (min): 86.5221
OOV Rate (% Averaged): 1.2065
1-gram Coverage (% Averaged): 91.9425
2-gram Coverage (% Averaged): 86.2483
3-gram Coverage (% Averaged): 59.4201
Vocabulary Size (Avg of Training) 96374
Total Words 8248002
RLAT – Text Collection and Selection – Snapshot function

• Snapshot functionality
  – Informative and visual feedback about the quality of text being crawled
  – Results which indicate the quality (PPL, OOV) of the collected texts are computed and displayed periodically (to be defined by the user) during the crawling process

Previous Language Model Performances on uploaded test set:

5. snapshot : November 16 2009 22:03:42: performance and crawled text and log file
15. snapshot : November 26 2009 22:03:44: performance and crawled text and log file
17. snapshot : November 28 2009 22:03:44: performance and crawled text and log file
18. snapshot : November 29 2009 22:03:45: performance and crawled text and log file
19. snapshot : November 30 2009 22:03:45: performance and crawled text and log file

Ngoc Thang Vu, Tim Schlippe, Franziska Kraus, and Tanja Schultz,
Rapid Bootstrapping of five Eastern European Languages using the Rapid Language Adaptation Toolkit,
Interspeech 2010.
RLAT – Text Collection and Selection – Snapshot function

Crawl

Every 10 mins

normalize

Every selected period

Copy, cat and build LM
RLAT – Rapid Language Adaptation Toolkit - 17

**Example:** [www.cnn.com](http://www.cnn.com) evaluated on English GlobalPhone dev set
Example: 18 days of Crawling French Data: www.leparisien.fr

**OOV Rate (%) of collected data on a French dev set**

**n-gram coverages (%) of collected data on a French dev set**
RLAT – Text Collection and Selection – Crawling

- Crawling of French Text Data: www.leparisien.fr

Vocabulary size of collected data

Total words of collected data
RLAT – Text Collection and Selection – Recrawling

- Boost data from similar sites

Approach

1. User supplies an URL to RLAT 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
Additional Data
We have found domain terms that might help get more text data relevant to the domain:

- **Check** those terms that are not relevant and hit the **Crawl_Again** button to get additional data

- Upload text file with search terms for additional data to build interpolated Language Model (**Note**: Each search term in one line):

  - **Durchsuchen..**
  - **Upload_AdditionalSearchTerms**

  - Number of hits from search engine: 1

- Calculate frequency statistics

  - **Calc_Stats** calculate word frequencies
  - **word frequencies**
1. Pure Rule-Based text normalization

2. Text normalization based on Statistical Machine Translation and user support
   - Pre-Normalization
     - Language-independent rule-based normalization / language-specific rule-based normalization (if available)
   - Language-specific normalization by RLAT users
     - Web-based user interface for text normalization
     - Keep the effort for the users low:
       - No use of sentences with more than 30 tokens to avoid horizontal scrolling
       - Sentences to normalize are displayed twice:
         The upper line shows the non-normalized sentence, the lower line is editable.

RLAT – Text Collection and Selection – Text Management

- Allow to store and handle multiple text files
RLAT – Text Collection and Selection – Text Management

www.tagesschau.de_2010-03-18_15.33.info

Date: 2010-03-18, 15:33
URL: http://www.tagesschau.de/
Maximal link depth: 1
Crawling language: German
Normalization language: German
Remain in web-domain: no
Size of crawled text: 696K
RLAT – Prompt Selection

• Prompts for recording:
  – Collection without transcription

• Prompts should be:
  – Easy to say (no hard words, numerals etc)
  – Rich in variability

( data_00001 "will we ever forget it." )
( data_00002 "there was a change now." )
( data_00003 "it fairly clubbed me into recognizing it." )
( data_00004 "i had faith in them." )
( data_00005 "he moved away as quietly as he had come." )
( data_00006 "suddenly his fingers closed tightly over the handkerchief." )
RLAT – Language Model Building

• Three LM building options
  – Do not use additional data
  – Add additional data: concatenate text corpora and build one LM
  – Interpolate base LMs

• Evaluation:
  – Perplexity, OOV Rate, Ngram Coverage, Vocabulary Size
  – 5-fold Cross Validation, Test File
RLAT – Language Model Building

Language Model Performance:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perplexity (Averaged)</td>
<td>27.6301</td>
</tr>
<tr>
<td>Perplexity (min)</td>
<td>26.8446</td>
</tr>
<tr>
<td>Perplexity (max)</td>
<td>30.0191</td>
</tr>
<tr>
<td>OOV Rate (% Averaged)</td>
<td>1.3308</td>
</tr>
<tr>
<td>1-gram Coverage (% Averaged)</td>
<td>98.85626</td>
</tr>
<tr>
<td>2-gram Coverage (% Averaged)</td>
<td>87.1453</td>
</tr>
<tr>
<td>3-gram Coverage (% Averaged)</td>
<td>55.63288</td>
</tr>
<tr>
<td>Vocabulary Size (Avg of Training)</td>
<td>1052</td>
</tr>
<tr>
<td>Total Words</td>
<td>26560</td>
</tr>
</tbody>
</table>

Interpolated Language Model Performance:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perplexity (Averaged)</td>
<td>30.63308</td>
</tr>
<tr>
<td>Perplexity (min)</td>
<td>28.2366</td>
</tr>
<tr>
<td>Perplexity (max)</td>
<td>33.5173</td>
</tr>
<tr>
<td>OOV Rate (% Averaged)</td>
<td>0.5566</td>
</tr>
<tr>
<td>1-gram Coverage (% Averaged)</td>
<td>99.52136</td>
</tr>
<tr>
<td>2-gram Coverage (% Averaged)</td>
<td>88.37218</td>
</tr>
<tr>
<td>3-gram Coverage (% Averaged)</td>
<td>55.9379</td>
</tr>
<tr>
<td>Vocabulary Size (Avg of Training)</td>
<td>53015</td>
</tr>
<tr>
<td>Total Words (Base+Recrawled)</td>
<td>914067</td>
</tr>
<tr>
<td>Weight (% Base Model)</td>
<td>98.90546</td>
</tr>
</tbody>
</table>

To build a language model, click this button:

Build Language Model

Select the n-gram order: 3

Select number of topics in your data: 0

Select the language model cutoffs: 2-gram: 0, 3-gram: 0, 4-gram: 0, 5-gram: 0

It appears you have collected additional data. You may want to:
- Build language model using this data and interpolate with main LM (recommended).
- Add this data to the original data and build a language model.
- Do not use additional data.

Upload a language model for (re) building the current (possible interpolated) LM interpolating it with the uploaded LM. The LM should be in plain text.

Upload Interpolation

Upload a Language Model (.lm or .tg) If there is already another LM build it will be replaced.
Notice that the upload limit is 30MB. Another possibility is to upload a clean text and build the language model.
RLAT – Language Model Building

- Interpolate LMs, LM Evaluation …

Upload two Language Models and one development set. These LMs will be mixed. The development set is used to calculate the weights. All files need to be text files. Caution: If there is already another LM build it will be replaced.
RLAT – Language Model Building

• More transparency for the LM building process

Language model

Build language model
(View language model file)
(View vocabulary file) (View language model build log file) (View language model input & result file)
Language model build complete

Your language model is built. If you want to rebuild it, your old language model will be backed-up at:
applications//German//lm-backup

Language Model Performance:

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perplexity (Averaged)</td>
<td>138.91328</td>
</tr>
<tr>
<td>Perplexity (min)</td>
<td>73.9213</td>
</tr>
<tr>
<td>Perplexity (max)</td>
<td>303.221</td>
</tr>
<tr>
<td>OOV Rate (% Averaged)</td>
<td>60.4339</td>
</tr>
<tr>
<td>1-gram Coverage (% Averaged)</td>
<td>46.76766</td>
</tr>
<tr>
<td>2-gram Coverage (% Averaged)</td>
<td>13.83348</td>
</tr>
<tr>
<td>3-gram Coverage (% Averaged)</td>
<td>2.064325</td>
</tr>
<tr>
<td>Vocabulary Size (Avg of Training)</td>
<td>103</td>
</tr>
<tr>
<td>Total Words</td>
<td>734</td>
</tr>
</tbody>
</table>
INPUT Parameters:
Text = default, n-gram order = 3, project language = Other.
n-gram cutoffs: 2-gram: 0, 3-gram: 0, 4-gram: 0, 5-gram: 0.

HIDDEN Parameters:
Discounting method: original (unmodified) Kneser-Ney discounting

RESULTS:
Original language model:

Cross validation:
Perplexity (Averaged) 55.65822
Perplexity (min) 37.8686
Perplexity (max) 85.1951
OOV Rate (% Averaged) 9.2900
1-gram Coverage (% Averaged) 91.93364
2-gram Coverage (% Averaged) 67.50566
3-gram Coverage (% Averaged) 46.44224
Vocabulary Size (Avg of Training) 7846
Total Words 86052
RLAT – Build Language Model – LM Management

- Build LMs from text selected in „Text Management“ and interpolate those.

```
crawledtext-current  □ optional weight
moredata-date_time  □ optional weight
upload-date_time    □ optional weight
```
RLAT – Build Language Model – LM Management

- Manually selected
- Dev set
- Weights
- Interpolate
- LM1
- LM2
- ...
- Text1
- Text2
- ...

LM Management
Language model management

(View LMMangement log file) (View LMMangement input & result file)
(View interpolate LMs log)

Select the ngram order:

Language models

If you provide language model weights, the sum of the selected language model weights should add up to 1. If the sum of the weight of the selected language models is smaller than 1, the weight of the second selected language model will be increased in a way that the weights of the selected language models add up to 1.

☑ www.tagesschau.de-2010-03-18_1501 www.tagesschau.de-2010-03-18_1501.info
☑ upload-cnnArt-2010-03-18_1511 upload-cnnArt-2010-03-18_1511.info
☑ upload-heiseArt-2010-03-18_1411 upload-heiseArt-2010-03-18_1411.info
☑ upload-zdfArt-2010-03-18_1527 upload-zdfArt-2010-03-18_1527.info

(View interpolated language model)

Select the language model cutoffs: 2-gram: 0 3-gram: 0 4-gram: 0 5-gram: 0

Delete selected language models.
Delete selected language models

Build language models from selected texts. Note that this function will overwrite old language models.
Build LMs from selected texts

Create an interpolated language model with user specified weights.
Create an interpolated language model with user specified weights.

Upload a development set. Note that this function will overwrite the interpolated language model. If language models, which are corresponding to the selected texts, do not exits, they will be build.
Upload dev set and interpolate language
Overview – Automatic Speech Recognition
RLAT – Phoneme Selection

- Selection from standard IPA chart
- User’s names for phonemes
  - Can match their lexicon (if one exists)
  - Can match their familiarity
- Audio feedback
  - Click to hear recording of each phone
**Phoneme set specification**

This is a tool which will display all IPA phonemes. You can choose and give names to phonemes you wish your models to use. After you have finished, click the "Submit" button.

**PhoneMapFile**

settings file

**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>Bilabial</th>
<th>Labiodental</th>
<th>Dental</th>
<th>Alveolar</th>
<th>Postalveolar</th>
<th>Retroflex</th>
<th>Palatal</th>
<th>Velar</th>
<th>Uvular</th>
<th>Pharyngeal</th>
<th>Glottal</th>
</tr>
</thead>
<tbody>
<tr>
<td>PH p</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B b</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T t</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DH d</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K k</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>g</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>+Q? +</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Nasal**

<table>
<thead>
<tr>
<th>m</th>
<th>n</th>
<th>mj</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>n</td>
<td>mj</td>
</tr>
</tbody>
</table>

**Trill**

<table>
<thead>
<tr>
<th>B</th>
<th>R</th>
</tr>
</thead>
</table>

**Tap or Flap**

<table>
<thead>
<tr>
<th>F</th>
<th>t</th>
<th>SH s</th>
<th>S s</th>
<th>SR s</th>
<th>C ç</th>
</tr>
</thead>
<tbody>
<tr>
<td>φ</td>
<td>f</td>
<td>θ</td>
<td>θ</td>
<td>θ</td>
<td>θ</td>
</tr>
<tr>
<td>F</td>
<td>f</td>
<td>H</td>
<td>H</td>
<td>H</td>
<td>H</td>
</tr>
</tbody>
</table>

**Fricative**

<table>
<thead>
<tr>
<th>Y</th>
<th>β</th>
<th>V v</th>
<th>DH δ</th>
<th>ZH h</th>
<th>ZH h</th>
</tr>
</thead>
</table>

**THE INTERNATIONAL PHONETIC ALPHABET (revised to 2005)**

**Consonants (Pulmonic)**

<table>
<thead>
<tr>
<th>Place</th>
<th>Bilabial</th>
<th>Labiodental</th>
<th>Dental</th>
<th>Alveolar</th>
<th>Postalveolar</th>
<th>Retroflex</th>
<th>Palatal</th>
<th>Velar</th>
<th>Uvular</th>
<th>Pharyngeal</th>
<th>Glottal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nasal</td>
<td>m n mj</td>
<td>t d</td>
<td>t d</td>
<td>d q c</td>
<td>j k g q g</td>
<td>j n</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trill</td>
<td>B R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tap or Flap</td>
<td>v f</td>
<td>θ s</td>
<td>θ s</td>
<td>s s</td>
<td>j x e h</td>
<td>h h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Vowels**

<table>
<thead>
<tr>
<th>Front</th>
<th>Central</th>
<th>Back</th>
</tr>
</thead>
<tbody>
<tr>
<td>i y</td>
<td>u o</td>
<td>å ö</td>
</tr>
<tr>
<td>ï ü</td>
<td>æ æ</td>
<td>è é</td>
</tr>
</tbody>
</table>

**Other symbols**

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>VowelIPA</td>
<td>VowelIPA</td>
<td>VowelIPA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>W</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>VowelMid</td>
<td>VowelMid</td>
<td>VowelMid</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Please choose the consonant sounds you'd like to have in your new acoustic models by checking the check box.
RLAT – Phoneme Selection

NOTE: If you do not specify a name for a diacritic phoneme, it will be ignored!

<table>
<thead>
<tr>
<th>Suprasegmentals:</th>
<th>a:</th>
<th>a: Long a · Half-long a · Extra-short</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syllabic:</td>
<td>ṛ ṛ Syllabic ṛ Non-syllabic</td>
<td></td>
</tr>
<tr>
<td>Consonant-release:</td>
<td>ṛ Aspirated ṛ Nasal release ṛ No audible release ṛ Lateral release</td>
<td></td>
</tr>
<tr>
<td>Phonation:</td>
<td>ṛ Voiceless ṛ Voiced ṛ Breathy voiced ṛ Creaky voiced</td>
<td></td>
</tr>
<tr>
<td>ṛ Dental ṛ Linguolabial ṛ Apical ṛ Laminal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Articulation:</td>
<td>ṛ Advanced ṛ Retracted ṛ Centralized ṛ Mid-centralized</td>
<td></td>
</tr>
<tr>
<td>ṛ Raised ṛ Lowered</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Co-articulation:</td>
<td>ṛ More rounded ṛ Less rounded ṛ Labialized ṛ Palatalized</td>
<td></td>
</tr>
<tr>
<td>ṛ Velarized ṛ Pharyngealized ṛ Retracted tongue root</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ṛ Nasalized ṛ Rhotacized</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Phoneme to dictionary assignment:

a: AL

### DIACRITICS

<table>
<thead>
<tr>
<th>DIACRITICS</th>
<th>Diacritics may be placed above a symbol with a descender, e.g. ŋ</th>
</tr>
</thead>
<tbody>
<tr>
<td>ṛ Voiceless</td>
<td>ṛ N̄ d̄</td>
</tr>
<tr>
<td>ṛ Voiced</td>
<td>ṛ B̄ ā</td>
</tr>
<tr>
<td>ṛ Creaky voiced</td>
<td>ṛ B̄ ā</td>
</tr>
<tr>
<td>ṛ Apical</td>
<td>ṛ T̄ d̄</td>
</tr>
<tr>
<td>ṛ Aspirated</td>
<td>ṛ T̄ d̄</td>
</tr>
<tr>
<td>ṛ Lateral release</td>
<td>ṛ T̄ d̄</td>
</tr>
<tr>
<td>ṛ Retracted tongue root</td>
<td>ṛ T̄ d̄</td>
</tr>
<tr>
<td>ṛ Nasalized</td>
<td>ṛ N̄ d̄</td>
</tr>
<tr>
<td>ṛ Rhotacized</td>
<td>ṛ R̄ t̄</td>
</tr>
</tbody>
</table>
RLAT – Lexicon Pronunciation Creation

- 3 options to obtain pronunciations:
  - Lex Learner
  - IPA Crawler (Web-derived pronunciations)
  - Keynounce – A game for pronunciation generation
RLAT – Lexicon Pronunciation Creation – G2P Rules

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 [Durchsuchen...]
Upload char.info [Durchsuchen... Upoload]

Download/view current settings: g2p  char.info

Example:

- aE [upper case] [lower case] [punctuation mark] [number] [others]
- b [upper case] [lower case] [punctuation mark] [number] [others]
- c [upper case] [lower case] [punctuation mark] [number] [others]
- d [upper case] [lower case] [punctuation mark] [number] [others]
- e [upper case] [lower case] [punctuation mark] [number] [others]
- f [upper case] [lower case] [punctuation mark] [number] [others]
- g [upper case] [lower case] [punctuation mark] [number] [others]
- hH [upper case] [lower case] [punctuation mark] [number] [others]
- iH [upper case] [lower case] [punctuation mark] [number] [others]

Total number of rules: 29
Thank you for giving us G2P rule!
Build Your System
- Text and prompt selection (help)
- Audio collection (help)
- Phoneme selection (help)
- Grapheme-to-phoneme rules (help)
- Lexicon pronunciation creation (help)
- Build acoustic model (help)
- Build language model (help)
- Test ASR system
- Create speech synthesis voice

User: demo Language: English Project: DemoSystem  [Logout]
Lexicon pronunciation creation

0% Prompt word coverage (0 of 310)

0% Corpus word coverage (0 of 1362)

0% Corpus token coverage

and

AE ND

listen to suggested pronunciation

Accept Pronunciation If this pronunciation acceptable.

Skip this word To work on it later.

Remove this word If this is not a valid word in your language.

Save and Build Lexicon Save your lexicon build LTS rules.

Lexlearner 8004

Phoneme labels for your language:
P B T D K G M N NG F V
TH DH S Z SHZH HH R
Y W L IY UW IH UH EY
ER OW AX EH AH AO
AE AA AY AW OY CH JH
2.1 Data – Wiktionary

- Wiktionary ([vikˈʃənəri], n):

- Das freie Wörterbuch:

  - Ein Wörterbuch:

    - Hauptsitz
    - Themenportale
    - Zufallige Seite
    - Inhaltsverzeichnis

- Eintrag:

  - Disziplin:
    - Sprache
    - Wissenschaft
    - Recht

- Hintergrund:

  - Das freie Wörterbuch:
    - Wörterbuch:
      - Deutsch
      - Französisch

- Hilfsverb:

  - Sein:
    - Präteritum: war, Partizip II: ge-wesen

- Aussprache:

  - IPA: [zaːn], [zən], [bist], [ist], sind: [zont], [zont], Präteritum: [vao], Partizip II: [ge-vənən]
Manually select in which Website to search for pronunciations

Our Automatic Dictionary Extraction Tool takes a vocabulary list with one word per line

For each word, the matching page is looked up (e.g. http://fr.wiktionary.org/wiki/abandonner)

If the page cannot be found, we iterate through all possible combinations of upper and lower case

Each web page is saved and parsed for IPA notations:

- Certain keywords in context of IPA notations help us to find the phonetic notation
  (e.g. \texttt{title="annexe:Prononciation/francais"} <span class="API" title="prononciation API"/>)
- For simplicity, we only use the first phonetic notation, if multiple candidates exist
- Our tool outputs the detected IPA notations for the input vocab list and reports back those words for which no pronunciation could be found
RLAT – Web-derived Pronunciations

Web derived pronunciations

This function allows you to extract pronunciations in IPA format from one or more websites. It is intended for Wiktionary or similar online dictionaries that have separate pages for individual words. Each source URL is queried with each word from a search vocabulary.

Source URLs

Please enter a list of URLs into the textbox below, one URL per line. Each URL must contain the placeholder "$$$" or "!!!". The former will be substituted by case variations of words from your search vocabulary, whereas the latter will be substituted by the true case of words exactly as written in your vocabulary.

http://de.wiktionary.org/wiki/$$$
http://de.wikipedia.org/wiki/$$$

Search vocabulary (word list)

Please upload a list of words whose IPA pronunciations you would like to find on the websites given by the source URLs. The upload must be a plain UTF-8 text file. It can be a vocabulary list (one word per line) or a reference text (several words per line). A list of unique words will be automatically extracted from the upload. However, no text normalization or filtering is performed, so you have to make sure that the upload contains only characters that are compatible with the specified source URLs.

Durchsuchen

Upload Data & Start WDP Crawl

Status

IPA crawler is running.
Searching URLs for word "Keyboard" (7 of 13) ...

53.85%

Result

(view vocab) (view log) (view wget log) (view output) (view generated dictionary)

Tim Schlippe, Sebastian Ochs, and Tanja Schultz,
Wiktionary as a Source for Automatic Pronunciation Extraction, Interspeech 2010.
Keynounce – A Game for Pronunciation Generation through Crowdsourcing

insurance

insuːrəns

< < del > >
Overview – Automatic Speech Recognition

[Diagram showing the process of Automatic Speech Recognition: AM, Lex, and LM with examples of spoken words.]

- AM: AM
- Lex: hi /h//ai/
you /j/u/
we /w//i/
- LM: hi you
you are
I am
RLAT – Audio Recording Tool

- Online recording tools
  - Collaboratively record large number of speakers
  - Speakers may separate from developers

- Java based for portability
  - Works with “many” browsers

- Visual feedback during recording

- Automatic upload on completion
RLAT – Audio Collection

The RLAT Toolkit provides tools for rapid language adaptation, including:

- Build Your System
  - Text and prompt selection (help)
  - Audio collection (help)
  - Phoneme selection (help)
  - Grapheme-to-phoneme rules (help)
  - Lexicon pronunciation creation (help)
  - Build acoustic model (help)
  - Build language model (help)
  - Test ASR system
  - Create speech synthesis voice

Audio collection

User: demo
Language: English
Project: DemoSystem

Record audio: [Watch Demo Video]

Current Status:
- Speaker ID:
- Speaker Name:
- Prompts Completed:

File: data_00001.wav
Length: 0.0
Position: 0.0

will we ever forget it.

Sessions Panel

- Set Working Dir
- New Speaker
- Close Speaker
- Play
- Record
- Upload

Process Log

1. SUCCESS: Server path set to demo/English/DemoSystem
RLAT – Acoustic Model Building

- Acoustic Model Building requires:
  - Recorded Speech Data
  - Phone set definition
  - Pronunciation Lexicon

- Two step process:
  1. Configuration
  2. Model Training
RLAT – 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
  - Train and test on selected sets
  - Few data: K-fold cross-validation, with K = #speakers
  - More data: Data split into 90% (train) and 10% (test)
RLAT – Acoustic Model Building

- Select speakers for test and training set

**Build acoustic model**

**Step 1. Janus Database Creation**

- Configure which speakers go into the different sets. The speaker config file which can be uploaded below, should contain the speaker numbers of all test speakers separated by space. For example 001 005 012 017 021. If you also want to specify dev speakers, write them into the second line of the file.

- Or you can let the system assign the speakers automatically to training set and test set.

  ![Button](Durchsuchen.png)  ![Button](Set Speakers and Configure Training.png)  ![Button](Configure Training.png)
RLAT – Acoustic Model Building – Training

• Requires successful configuration

• Creates Log files and Displays to the user
  – All steps of training
    • 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 (Mel Frequency Cepstral Coefficients), LDA (linear discriminant analysis)

• Progress of training procedure
• Results of performance evaluation
RLAT – Acoustic Model Building – Training

Step 2. Acoustic Model Training

- Select method of acoustic model training.
  1. Either the specified speakers (see below) or 10% random speakers will be selected for tuning (devSet) and evaluating (testSet) the performance of the system. The remaining speakers will be selected for training the acoustic models.
  2. 90% of each speaker’s utterances are used for training and the rest of the utterances are used for testing the acoustic models.
  3. All available data will be used for training the acoustic models. Testing can be done via the ‘Test ASR system’ interface.

Train_Acoustic_Models LOG_FILE

- Initialization: IN PROGRESS...
- Computing Labels: NOT STARTED YET.
- Computing Cepstral Means: NOT STARTED YET.
- Computing LDA matrix: NOT STARTED YET.
- K-Means Clustering over the codebooks: NOT STARTED YET.
- EM Training: NOT STARTED YET.
- Segmentation of Speech: NOT STARTED YET.
- Making Polyphone Trees: NOT STARTED YET.
- EM Training over the code-books: NOT STARTED YET.
- Clustering Contexts: NOT STARTED YET.
- Performing the Splits: NOT STARTED YET.
- Context Dependent Acoustic Models Training: NOT STARTED YET.

View/Hide Results >>

Step 3. Acoustic Model Testing

Currently selected Language Model for testing: Default LM - See page ‘Build language model’

- Decode on specified test set.
- Decode on specified development set.

Test_Acoustic_Models
Overview – Automatic Speech Recognition

AM | Lex | LM
---|-----|---
hi /h//ai/ | you /j/u/ | hi you you are
we /w//i/ | I am |
Simple echo back testing function
Thanks for your interest!
References


