LEXICON MODELING

Dr. Ivan Kraljevski, Dr. Frank Duckhorn

Fraunhofer Institute for Ceramic Technologies and Systems IKTS, Dresden

Daniel Sobe

Stiftung für das sorbische Volk, Bautzen

Table of contents

1 Introduction

Description: The aim of this work package is to improve the modeling of the pronunciation of Upper Sorbian words. The modeling is implemented with grapheme-to-phoneme (G2P) models.
These are supposed to improve the pronunciation variants and represent the words better.

There are two main approaches:

1. To use a large existing lexicon (min. 300,000 words) with pronunciation variants for the Upper Sorbian to train a G2P models.
2. To improve and extend the rule-based, manually created G2P mappings from the feasibility study. For this, expert knowledge of a phonetician/linguist is required. Together with the expert(s) the rules for canonical pronunciation of Upper Sorbian words will be re-defined and existing rules will be enhanced by further pronunciation variants.

Deliverables: Software to add new words to an existing lexicon and to generate pronunciation variants for them, a short report.

Impact: Adding new pronunciation variants of words that are in the existing language model (Smart Lamp), thus generating an improved speech recognition configuration. Together with work-package 3 (WP3) additional words can be added to existing applications (SmartLamp).

Prerequisites:

- large lexicon with at least 300,000 words or
- support by phonetician/linguist

2 Lexical modeling

Lexical modeling is done by employing simple G2P rules enhanced with an extended list of exception rules. The lexicon for a given textual corpus is automatically created by the BASGenerator script as described in the work package (WP1). The generated lexicon was used for acoustic modelling and for speech recognition.

The provided lexicon (manually created) in Upper Sorbian contains around 4000 words, hence it not suitable for reliable statistical pronunciation modelling using Finite-State-Transducers (FST). The provided lexicon is not peer reviewed by experts and hence do not meet the prerequisites of the WP2.

We made a basic comparative analysis of both approaches, and the step-by-step guide is provided to allow future possibility for statistical G2P modeling.

Table 1 presents a small selection of words with the manually created pronunciations in IPA format, their SAMPA version and the automatically pronunciations generated by the rules with the BASGenerator script.

In this case, around 25% of the automatically generated are matching the manually created pronunciations by a phonetician.

Table 1. Comparison of manually and automatically created pronunciations.

WORD	IPA	SAMPA	AUTOMATIC G2P (UASR)	MATCH
agreement	ɛˈgrimɛnt	E g r i m E n t	a g r E m E n t	no
aktiwěrować	aktiˈu̯ɪrou̯aʧ	a k t i w I r o w a tS	a k t i u ji r o w a tS	no
alawn	alaun	a l a u n	a l a u n	yes
alowej	aloei̯	a l o e i	a l o w e j	no
amen	amɛn	a m E n	a m E n	yes
amizantny	amiˈzantnɨ	a m i z a n t n 1	a m i z a n t n 1	yes
amizěrować	amiˈzɪrou̯aʧ	a m i z I r o w a tS	a m i z ji r o w a tS	no
angorawołma	aŋˈgɔrau̯ɔma	a n g O r a w O m a	a n g O r a w o u m a	np

3 Statistical G2P modeling

Some of common tools used for grapheme to phoneme modelling in speech recognition toolkits are Phonetisaurus[1] and Sequitur G2P[2].

The first one is based on an N-gram based translation models and usually implemented as a weighted finite state transducer (WFST).

Recently Long-Short-Time-Memory (LSTM) Recurrent Neural Networks (RNN) and Transformer based G2P models are becoming more popular. In both approaches, carefully prepared training lexicon is necessary with as many as possible, relevant pronunciation variants.

If such training lexicon is available, we suggest using the Python library phonetisaurus-pypi[3], as a simple and user-friendly open-source tool.

The procedure using the original phonetisaurus toolkit is rather complicated and involves installation of third-party software with restricted licenses (MITLM[4]).

3.1 Grapheme-2-phoneme training

The training is executed by the python script, given the assumptions that there is a lexicon in the format `[word phoneme phoneme …]`:

```
phonetisaurus train --model models/g2p.fst sources/lexicon.lex
```

[1] https://github.com/AdolfVonKleist/Phonetisaurus

[2] https://www-i6.informatik.rwth-aachen.de/web/Software/g2p.html

[3] https://github.com/rhasspy/phonetisaurus-pypi

[4] https://github.com/mitlm/mitlm

3.2 Grapheme-2-phoneme prediction

The created model can be used to generate new pronunciations of words unseen in the training with the option to provide the N-best candidates.

```
phonetisaurus predict --model=models/g2p.fst --nbest=5 ĆOPŁOBĚłU

ĆOPŁOBĚłU tS O p w O b I w u
ĆOPŁOBĚłU tS O p w O b I u
ĆOPŁOBĚłU tS O p w O b e u
ĆOPŁOBĚłU tS O p w O b e i u
ĆOPŁOBĚłU tS U p w O b I w u
```

Or in the case where there is a vocabulary or list of words for which the pronunciations should be generated:

```
phonetisaurus predict --model=/models/g2p.fst --nbest=3 < sources/hsb.vocab > results/hsb_gp.lex
```

It is important to use consistent character casing, the words' case in the lexicon used for training should match the words used for prediction (lower or upper case).

4 Deliveries

The following data is delivered in this work package:

- Excel file with the comparison table of the manually and automatically created pronunciations.
- Source lexicon in X-SAMPA format (lexicon.lex).
- Created g2p model (g2p.fst).
- Vocabulary for prediction (hsb.vocab).

Part of this work package deliveries are included into the WP4:

- Phoneme inventory
- Exception rules
- Improved rule based G2P generator script (BASGenerator)