Appendix
Language Modeling for Code-Switching: Evaluation, Integration of Monolingual Data, and Discriminative Training

Hila Gonen[1] and **Yoav Goldberg**[1,2]
[1]Department of Computer Science, Bar-Ilan University
[2]Allen Institute for Artificial Intelligence
{hilagnn,yoav.goldberg}@gmail.com

A Appendix

A.1 Creating Evaluation Dataset – Implementation Details

Our implementation is based on the Carmel FST toolkit.[1] We create an FST for converting a sentence into a sequence of phonemes, and its inverse FST. The words to phoneme mapping is based on pronunciation dictionaries, according to the language tag of each word in the sentence.

We use The CMU Pronouncing Dictionary[2] for English and a dictionary from CMUSphinx[3] for Spanish. As the phoneme inventories in the two datasets do not match, we map the Spanish phonemes to the CMU dict inventory using a manually constructed mapping.[4]

To favor frequent words over infrequent ones, we add unigram probabilities to the edges of the transducer (taken from googlebooks unigrams[5]). We filter some words that produce noise (for example, single letter words that are too frequent). When creating a monolingual sentence, we use an FST with the words of that language only.

As many phoneme sequences in Spanish do not produce English alternatives (and vice versa) we allow minor changes in the phoneme sequences between the languages. Specifically, we create a small list of similar phonemes (such as "B" and "V"),[6] and generate an FST that for each phoneme allows changing it to one of its alternatives or dropping it with low probability.

Since using the whole sentence has higher chances of encountering words that are not included in the dictionaries, we only convert a sampled part of the gold sentence when creating a code-switched alternative. This also results in alternatives with higher similarity to the gold sentence. However, when creating a monolingual alternative (i.e. a Spanish alternative to an English gold sentence), we have no choice but to use the whole sentence.

A.2 Data

Code-switching corpus We pre-processed the Bangor Miami Corpus by lower-casing it and tokenizing using the spaCy tokenizer.[7] We did not reduce the vocabulary size which was quite small to begin with (13,914 words). After preprocessing, we got 45,621 sentences with 322,044 tokens.

Monolingual corpora This data from the OpenSubtitles2018 corpus (Tiedemann, 2009) comes pre-tokenized. We pre-processed it by lower-casing, removing parenthesis and their contents, and removing hyphens from beginning of sentences.

We use 1M lines from each language, resulting in 7,501,714 tokens in English and 6,566,337 tokens in Spanish. We have 45,280 words in the English vocabulary and 50K words in the Spanish one (reduced from 83,615).

A.3 Architecture and Training Details

The LSTM has a hidden layer of dimension 650. The input embeddings are of dimension 300. We

[1] https://www.isi.edu/licensed-sw/carmel/

[2] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[3] https://sourceforge.net/projects/cmusphinx/files/Acoustic%20and%20Language%20Models/Spanish/

[4] The full mapping from Spanish to English: ch-CH, rr-R, gn-NG, a-AA, b-B, b-V, e-EY, d-D, d-DH, g-G, f-F, i-IY, k-K, j-H, m-M, n-N, l-L, o-OW, p-P, s-S, r-R, u-UW, t-T, y-Y, x-S, x-SH, x-K S, x-H, z-TH, z-S, ll-L Y, ll-SH. We thank Kyle Gorman for helping with the mapping.

[5] http://storage.googleapis.com/books/ngrams/books/datasetsv2.html

[6] The full list of similar phonemes: OW - UW, AA - EY, L - M, N - M, M - L, B - P, B - V, V - F, T - D, K - G, S - Z, S - TH, Z - TH, SH - ZH

[7] https://spacy.io

use auto-batching with batches of size 20. We optimize with SGD and learning rate of 10, reducing it by a factor of 2.5 at the end of each epoch with no improvement. We also use clipping gradients of 1, and weight decay of 10^{-5}. We initialize the parameters of the LSTM to be in the range of $[-0.05, 0.05]$. We also use word dropout with the rate of 0.2. We set the dropout in our LSTM (Gal and Ghahramani, 2016) to 0.35. We train for 40 epochs and use the best model on the dev set.

References

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of NIPS*.

Jörg Tiedemann. 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, pages 237–248.