MAIS 202 Deliverable 1: Data Selection Proposal

Michelle Wang

1 Dataset

The ability to determine the pronunciation of a word based on its written form is very important in some fields like text-to-speech synthesis. This can prove to be a challenging task for some languages like English because they have highly irregular pronunciation rules. The goal of my final project will be to create a model that converts an English word to its corresponding sequence of phonemes. To do so, I will be using the CMU Pronouncing Dictionary [1]. This dataset contains more than 134 000 words and their North American English pronunciations, and it is often used to test the performance of grapheme-to-phoneme models.

2 Methodology

2.1 Data preprocessing

The dataset is distributed as a text file, with each line consisting of an English word followed by its pronunciation represented by ARPAbet symbols. I will first parse the dictionary to get the words and their pronunciation, skipping any word that contains characters that are not alphabet letters. Then I will use one-hot encoding to convert all of the words and pronunciation to a binary format.

2.2 Machine learning model

I will use a sequence-to-sequence model with an encoder and a decoder. This type of model is useful when the length of the input differs from the length of the output, which will be the case here since the mapping from grapheme to phoneme in a word is very often not one-to-one. I am considering implementing a model similar to the deep bidirectional Long Short-Term Memory (LSTM) model described by Rao et al. [2]. I would like to use LSTMs because they can make predictions based on contextual information and because they do not require explicit alignment of graphemes and phonemes. However, one potential drawback to using LSTMs is that they can be computationally costly and might overfit easily because they have a lot of parameters. Another type of neural network that I can use as encoder and/or decoder would be a Convolutional Neural Network (CNN), following an approach outlined by Yolchuyeva et al. [3].

2.3 Final conceptualization

My goal is to produce a simple landing-page web application that takes in a word and outputs its pronunciation encoded using the ARPAbet system. If time permits, it would also be interesting to include sound output that corresponds to the predicted pronunciation, since ARPAbet can be hard to decipher.

References

[1] Carnegie Mellon University. *The CMU Pronouncing Dictionary*. `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`.

[2] K. Rao et al. "Grapheme-to-phoneme conversion using Long Short-Term Memory recurrent neural networks". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Apr. 2015, pp. 4225–4229. DOI: `10.1109/ICASSP.2015.7178767`.

[3] Sevinj Yolchuyeva, Géza Németh, and Bálint Gyires-Tóth. *Grapheme-to-Phoneme Conversion with Convolutional Neural Networks*. Mar. 2019. URL: `https://www.mdpi.com/2076-3417/9/6/1143/htm`.