Generative Adversarial Phonology: Neural networks and unsupervised phonetic and phonological learning

Most of the existing computational models in phonology (both the MaxEnt model and more recent proposals using neural network architecture) model learning as symbol manipulation and operate with discrete units. Phonological learning is thus modeled as if phonetic learning had already taken place: the initial state already includes phonemes or feature matrices as discrete units that had already been learned.

This paper proposes that unsupervised phonetic and phonological learning of acoustic speech data can be modeled with Generative Adversarial Networks (GAN). The paper argues that GANs are uniquely appropriate for modeling phonetic and phonological learning because the network is trained on unannotated raw acoustic data, learning is unsupervised without any language-specific inputs, and the result is a network that learns to generate acoustic speech signal from random input variables. The main characteristic of GANs is the architecture that involves two networks: the Generator network and the Discriminator network (Goodfellow et al., 2014). The Generator network is trained to generate data from random noise, while the Discriminator is trained on distinguishing real data from outputs of the Generator network. The Generator is trained to generate data that minimizes accuracy of the Discriminator network. The training results in a Generator (G) network that takes random noise as its input (e.g. multiple variables with uniform distributions) and outputs data such that the Discriminator is inaccurate in distinguishing the generated from real data.

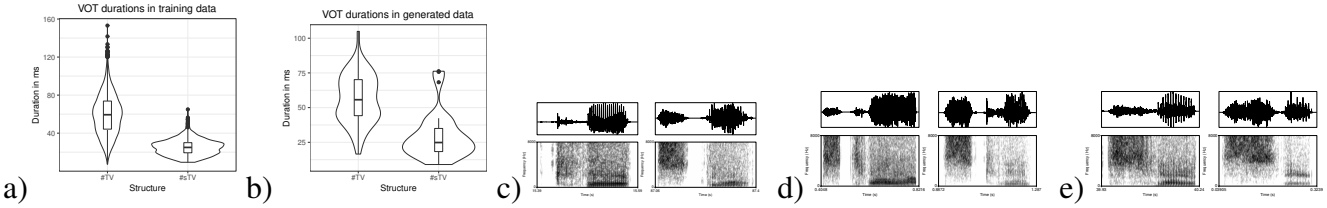
A GAN model for acoustic data proposed by Donahue et al. (2019) was trained on an allophonic distribution in English, where voiceless stops surface as aspirated word-initially before stressed vowels except if followed by a sibilant [s]. The corresponding sequences of word-initial voiceless stops followed by a vowel with and without the preceding [s] from the TIMIT database were used in training. The training data includes 4,930 sequences with the structure #TV (T = /p, t, k/) and 533 sequences with the structure #sTV (5,463 total). VOT in the training data is approximately 32.4 ms shorter [-34.3 ms, -30.6 ms] if T is preceded by [s] ($t = -34.16, p < 0.0001$).

The GAN architecture in Donahue et al. (2019) for audio data is based on Deep Convolutional Generative Adversarial Network proposed by Radford et al. (2015). The Generator network takes as input z, a vector of one hundred uniformly distributed variables and outputs 16,384 data points, which constitutes the output audio signal. The network has five 1D convolutional layers (Donahue et al., 2019). The Discriminator network takes 16,384 data points (raw audio file) as its input and outputs a single logit. The WaveGAN model (Donahue et al., 2019) uses ReLU activation in all but the last layer for the Generator network, and Leaky ReLU in all layers in the Discriminator network (as recommended for DCGAN in Radford et al. 2015).

The Generator network after 12,225 steps ($\sim$ 716 epochs) generates speech signal that appears close to actual acoustic data. To test whether the Generator learns the conditional distribution of VOT duration, the Generated samples were annotated for VOT duration. Altogether 96 generated samples were annotated, 62 in which no period of frication of [s] preceded and 34 in which [s] precedes the TV sequence. Absence of [s] is a significant predictor of VOT duration: $F(1) = 53.1, p < 0.0001$. The estimates for Intercept (duration of VOT when no [s] precedes) are $\beta = 56.2$ ms, $t = 25.74, p < 0.0001$. VOT is on average 26.8 ms shorter if [s] precedes the TV sequence ($\beta = -26.8$ ms, $t = -7.29, p < 0.0001$).

While VOT duration is significantly shorter if [s] precedes the #TV sequence in the generated data, the model shows clear traces that the learning is not complete and that the generator network is in the process of learning the distribution categorically at 12,255 steps. The three longest VOT durations in the #sTV sequences in the generated data are 68.3 s, 75.7 s, and 76.2 s. In all three

cases is the VOT longer than the longest VOT duration of any #sTV sequence in the training data (longest is 65 ms; Figure(a)). Figure (d) shows two such cases.



a) b) c) d) e)

Longer VOT duration in the #sTV condition in the generated data compared to training data is not the only violation of the training data that the Generator outputs and that resembles linguistic behavior in humans. Occasionally, the Generator outputs a linguistically valid #sV sequence for which no evidence was available in the training data. The minimal duration of closure in #sTV sequences in the training data is 9.2 ms, the minimal duration of VOT is 9.4 ms. All sequences containing [s] from the training data were manually inspected and none of them contain a #sV sequence without a period of closure and VOT. Figure (e) shows two cases of the Generator network outputting a #sV sequence without any stop-like fall of the amplitude. In other words, the Generator network outputs a linguistically valid sequence #sV without any evidence for existence of this sequence in the training data, which likely means that the network is treating [s] as a discrete unit that can be combined with other units. Such outcomes also suggest the trained model is not overfitted.

Establishing what neural networks learn is a challenging task. This paper argues that the Generator network shows traces of learning that can be directly linked to phonological features. We identify values in the latent space (uniformly distributed random variables) that are fed to the Generator that correspond to the presence of [s] in the output. The features are identified from a logistic regression model in which the hand-annotated presence or absence of [s] in generated data is the dependent variable and random input variables of the Generator network are the predictors. In other words, by setting 9 out 100 of variables in the latent space to a certain value, the Generator outputs almost exclusively #sTV sequences. These values in the input of the Generator network can thus be directly paralleled to features values (presence of [s]) in phonological theory.

Generated outputs from the Generator network replicate the conditional distribution of VOT duration in the training data. The Generator network thus not only learns to output signal that resembles human speech from noise (input variables sampled from a uniform distribution), but also learns to output shorter VOT durations when [s] is present in the signal. While this distribution is phonologically local, it is non-local in phonetic terms as a period of closure necessarily intervenes between [s] and VOT. In addition, the network produces innovative outputs that are linguistically legal, showing evidence that it treats segments such as [s] as discrete units that can be combined with other units in innovative ways.

These results suggest that the phonetic and phonological learning (of at least non-contrastive allophonic distributions) can be modeled with Generative Adversarial Neural networks. The advantage of this proposal is that learning is modeled from raw phonetic inputs with no language-specific inputs. GAN's architecture resembles the production-perception loop in speech: production is modeled with the Generator network, perception with the Discriminator network. Finally, the paper suggest that the learned input-output (latent variables-acoustic output) mapping involves variables that can be directly paralleled to phonological features. Several further applications of the model are discussed.