Generative Adversarial Phonology: Modeling unsupervised allophonic learning with neural networks

Gašper Beguš
University of Washington
begus@uw.edu

May 28, 2019

Abstract

This paper proposes that unsupervised phonetic and phonological learning of acoustic speech data can be modeled with Generative Adversarial Networks. Generative Adversarial Networks 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. A GAN model for acoustic data proposed by Donahue et al. (2019) was trained on an allophonic alternation 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 with and without the preceding [s] from the TIMIT database were used in training. Measurements of VOT of stops produced by the Generator network was used as a test of learning. The model successfully learned the allophonic alternation without any language-specific input: the generated speech signal contains the conditional distribution of VOT duration. The results demonstrate that Generative Adversarial Networks bear potential for modeling phonetic and phonological learning as they can successfully learn to generate allophonic distribution from only acoustic inputs without any language-specific features in the model. The paper also discusses how the model's architecture can resemble linguistic behavior in language acquisition.

keywords: artificial intelligence, neural networks, generative adversarial networks, phonetic learning, phonological learning, voice onset time, allophonic distribution

1 Introduction

How to model language acquisition is among the central questions in linguistics and cognitive science in general. Acoustic speech signal is the main input for infants acquiring language. By the time acquisition is complete, humans are able to decode and encode information from or to a continuous speech stream and construct grammar that enables them to do so. In addition to syntactic, morphological, and semantic representation, the learner needs to learn phonetics and phonology: to analyze and in turn produce speech as a continuous acoustic stream composed of discrete mental units called phonemes. Phonological grammar manipulates these discrete units and derives surface forms from stored lexical representations.

Computational models have been invoked for the purpose of modeling language acquisition ever since the rise of computational methods and computationally informed linguistics (for an overview, see Jarosz 2019; Pater 2019). Modeling phonetic and phonological learning is an inherently complex task: the ideal model would need to learn articulatory representations from unannotated acoustic inputs on the phonetic level together with underlying representations and derivations on the phonological level.

For example, phonemes are abstract discrete mental units, the smallest meaning-distinguishing units of language (Dell et al., 1993; Kawamoto et al., 2015). A string of phoneme constitutes a morpheme, the smallest meaning-bearing unit. Phonemes are represented as feature matrices: sets of binary contrastive features. For example, the phoneme /p/ is represented as [−sonorant, −continuant, +labial, −voice], which uniquely selects the phoneme /p/ from the inventory of English phonemes. This abstract unit can surface with variations on the phonetic level. English /p/ is realized as aspirated [p^{h}] word-initially before stressed vowels, but as unaspirated plain [p] if [s] immediately precedes it. This distribution is completely predictable and derivable with a simple rule Iverson and Salmons (1995), which is why the phoneme as an abstract mental unit is unspecified for aspiration (or absence thereof) in the underlying representation. Aspiration is represented as feature [±spread glottis]. A simple rule of the SPE-type phonology (Chomsky and Halle, 1968) in 1 can derive surface forms with our without the aspiration from underlying representation.[1]

$$\begin{bmatrix} -sonorant \\ -continuant \\ -voice \end{bmatrix} \rightarrow \begin{bmatrix} +spread\ glottis \end{bmatrix} /\#\_\_\_\_ \begin{bmatrix} +stress \end{bmatrix} \tag{1}$$

For example, lexically stored input strings of phonemes such as /ˈpɪt/ ‘pit’ and /ˈspɪt/ ‘spit’ are unspecified for aspiration. The rule in 1 loops over the input strings and assigns [+spread glottis] value if the condition #__[+stress] is met (i.e. when a segment, represented with an underline, is immediately preceded by a word boundary # and followed by [+stress]). The outputs are [ˈp$^{\mathbf{h}}$ɪt] with the aspiration and [ˈspɪt] without the aspiration. Table 1 illustrates the derivation.

Phonetically, this rule is explained in Kim (1970). The spreading of the glottis onsets during [s] in sT clusters. By the time the stop is released, the glottis contracts and the aspiration ceases, which results in absence of aspiration after the release of the stop. Even if phonologically, the rule in 1 is the consequence of both /s/ and /p, t, k/ being underlyingly [+spread glottis], which is why in the cluster sT the feature is not realized on the stop (Iverson and Salmons, 1995), the learner still needs to acquire this allophonic distribution from speech signal. The primary evidence for this allophonic distribution is VOT duration.

Input	/ˈpɪt/	/ˈspɪt/	
Derivation	ˈp^{h}ɪt	∅	aspiration rule
Output	[ˈp^{h}ɪt]	[ˈspɪt]	

Table 1: Derivation of /ˈpɪt/ and /ˈspɪt/ in the rule-based approach.

The main objection against the rule-based approach to phonology is that rules are too powerful and overgenerate. In other words, rule-based phonology can derive any output from a given input by applying multiple ordered rules in the derivation (e.g. a set of simple ordered rules can turn an input /ˈpit/ into the output [ˈʒk’æŋ] and infinite other outputs). Phonological typology, on the other hand, is considerably more limited. Moreover, modeling learning and phonological variation within the rule-based approach faces some crucial challenges (overview of the discussion in Albright and Hayes 2011; Heinz 2011).

[1] The account of aspiration in English is simplified for the purpose of this paper, because the model is trained on simplified conditions. For further details, see Iverson and Salmons (1995).

	/ˈpɪt/	*#[−spread glottis][+stress] $w = 2$	Ident-IO $w = 1.5$	H
	[ˈpɪt]	−1		−2
☞	[ˈpʰɪt]		−1	−1.5

Table 2: A tableau illustrating output-input optimization in Harmonic Grammar.

As a response to the rule-based approach and the problem of learnability and overgeneration, Optimality Theory (Prince and Smolensky, 1993/2004) and related proposals such as Harmonic Grammar and Maximum Entropy (MaxEnt) grammar (Legendre et al., 1990; Goldwater and Johnson, 2003; Legendre et al., 2006; Wilson, 2006; Hayes and Wilson, 2008; Pater, 2009; Hayes and White, 2013; White, 2014, 2017) were proposed. These models were heavily influenced by the early advances in neural network research (Pater, 2019). The main advantage of Optimality Theoretic architecture is that phonological computation is modeled as optimization of outputs based on inputs. Optimality Theory introduces constraints: functions that evaluate outputs or input-output pairs. Any given input has a set of potential outputs. The winning output is chosen based on constraint violations: the output that violates the lowest-weighted constraints is the winning candidate. For example, instead of deriving outputs from the input via rules, output [ˈpʰɪt] is chosen over a competing candidate [ˈpɪt] (for input /ˈpɪt/) because it satisfies the constraint stating that word-initial sequences of #[−sonorant, −continuant, −voice, −spread glottis] [+stress] are dispreferred. On the other hand, output [ˈspɪt] is chosen over a competing candidate [ˈspʰɪt] (for input /ˈspɪt/), because winning candidates tend to replicate inputs (the so-called faithfulness constraints, marked as Ident-IO). The output-input optimization is formalized via Harmony scores (H). Constraints are functions that assign negative integers if an output or input-output pair incurs a violation. Each constraint (C_i) has a weight (w_i). Harmony scores of output-input pairs (H(output, input)) are calculated as a sum of the product of constraint violations and their corresponding weights (Equation 2). The output candidate with the highest score is chosen as the winner (marked with ☞). Table 2 illustrates calculation of harmony scores based on constraint violations and their corresponding weights.

$$H(y, x) = \sum_{i=1}^{m} w_i C_i(y, x), \qquad (2)$$
$$\text{where } y = \text{output and } x = \text{input}$$

In other words, phonological computation is modeled as input-output mapping based on two competing forces (formalized as constraints): the tendency to satisfy some surface form requirement and the tendency to be faithful (as identical as possible) to the input. Unlike rule-based approach, Optimality Theory is substantially more restrictive: some processes are predicted to be unattested. The second advantage of Optimality Theoretic approaches to phonology is that the model provides a theory of learnability and derives non-categorical processes (phonological variation). Harmony scores can be transformed into probability distributions ($\mathrm{P}(y|x)$) over output candidates. In other words, every output candidate is assigned some probability of surfacing as the output, directly derivable from Equation 3 (Goldwater and Johnson, 2003).

$$P(y|x) = \frac{e^{H(x,y)}}{\sum\limits_{y \in Y(x)} e^{H(x,y)}}, \qquad (3)$$
$$\text{where } y = \text{output and } x = \text{input}$$

In the most standard version of MaxEnt and Optimality Theoretic approaches to phonology, constraints are predetermined in language acquisition (or at least constraint template that can be filled with feature matrices is; Hayes 1999). The main task of the learner is thus to learn constraint weights. This problem is computationally most successfully addressed within the Maximum entropy model (or a multinomial logistic regression with constraints as predictors) approach. The implementation, first proposed by Goldwater and Johnson (2003) has seen success in deriving phonological learning and gradient phenomena in phonology. Learning constraint weights is thus an optimization problem that can be solved with a Stochastic Gradient Descent or other appropriate algorithms for the task Pater (2019). Several works directly compare and parallel Maximum entropy grammar with experimentally observed human behavior (Wilson, 2006; White, 2014, 2017; Moreton et al., 2017). Another advantage of this model is that learning biases and asymmetries in rate of learning can be encoded in the computational model. Constraints can have non-zero prior weights and learning rate can be encoded as prior variance Wilson (2006) or prior means White (2017) in regularization term.

As noted by Pater (2019), the weighted-constraint approaches to phonology including Maximum entropy grammar approach are in many ways related to neural networks. Modeling linguistic data with neural networks has seen a rapid increase in the past few years (Avcu et al. 2017; Mahalunkar and Kelleher 2018; Weber et al. 2018; Prickett et al. 2019, for cautionary notes, see Rawski and Heinz 2019). While the Maximum entropy grammar as well as the rule-based approaches require language-specific devices (such as constraints or rules, binary features, discrete mental units of representation etc.), one of the promising implications of the neural network modeling is the ability to test generalizations that models produce without language-specific devices (Pater, 2019).

The existing computational models in phonology (both using the MaxEnt and neural network methods), however, model learning as symbol manipulation and operate with discrete units, either completely abstract made-up units or phonology-specific units called phonemes. In other words, most of the models already assume some level of abstraction and model learning as symbol manipulation either at the segmental level or operate with feature matrices. Phonological learning is thus modeled as if phonetic learning had already taken place: the initial state already includes phonemic inventories, phonemes as discrete units, or feature matrices that had already been learned.

One of the few models that operates with raw phonetic data, however, does not involve neural network architecture. Schatz et al. (2019) propose a Dirichlet process Gaussian mixture model that learns categories from raw acoustic input in an unsupervised learning task. The model is trained on English and Japanese data and the authors show that the asymmetry in perceptual [l]∼[r] distinction between English and Japanese falls out automatically from their model. The primary purpose of the model in Schatz et al. (2019) is modeling perception and categorization: they model computationally how a learner is able to categorize raw acoustic data into sets of discrete categorical units that have phonetic values (i.e. phonemes). Other proposals for unsupervised acoustic analysis with neural network architecture do not model phonetic and phonological learning, but are primarily concerned with unsupervised feature extraction.

This paper proposes that learning of phonetic and phonological processes can be modeled from raw acoustic inputs without any prior assumptions about discrete simbols with Generative Adversarial Networks (GAN). The primary purpose of the present paper is to model learning of a non-local allophonic distribution from raw acoustic data: A GAN model produces raw acoustic outputs from random noise based solely on acoustic training data. To the author's knowledge, this is the first proposal that uses GAN architecture to model generative phonetic and phonological learning.

A Generative Adversarial Network architecture implemented for audio files in Donahue et al. (2019) (WaveGAN) was trained on continuous raw speech data that contains information for an allophonic distribution: word-initial pre-vocalic aspiration of voiceless stops (['p^{h}ɪt] ∼ ['spɪt]). The data is curated in order to control non-desired effects, which is why only sequences of the shape #TV and #sTV are fed to the model. This allophonic distribution is uniquely appropriate for testing learnability in a GAN setting, because the dependency between the presence of [s] and duration of VOT is not strictly local. To be sure, the dependency is local in phonological terms, as [s] and T are two segments and immediate neighbors, but in phonetic terms, a period of closure intervenes between the aspiration and the period (or absence thereof) of frication noise of [s].

The advantage of the GAN architecture is that learning is completely unsupervised and that phonetic learning is simultaneous with phonological learning. There exist a vast literature on the relationship between phonetics and phonology. The discussion is highly complex and the purpose of this paper is not to solve it, but phonetics and phonology cannot be completely dissociated from each other. A network that models learning of phonetics from raw data and shows signs of learning discrete phonological units is one step closer to reality than a model that operates with symbolic computation and assumes phonetic learning had already taken place and is independent of phonology and vice versa. Additionally, GAN architecture models the production-perception loop in phonetics and phonology that other models limited to symbolic computation fail to do. Generator's outputs can be interpreted as the basis for articulatory targets in human speech. The latent variables in the input of the Generator can be modeled as articulatory parameters that the Generator learns to output into a speech signal by attempting to maximize the error rate of a Discriminator network that distinguishes between real data and generated outputs. The GAN network thus incorporates both the articulatory element (the Generator) as well as the perceptual element (the Discriminator) in speech production.

The hypothesis of the computational experiment presented in Section 3 is the following: if VOT duration is conditioned on the presence of [s] (i.e. there is significant difference between the two groups) in output data generated from noise by the Generator network, it means that the Generator network has successfully learned a phonetically non-local allophonic distribution. This distribution is not automatic in English, which means that not only phonetic, but also phonological distributions are modeled with this approach. The results suggest that phonetic and phonological learning can be modeled simultaneously and in unsupervised mode directly from what language acquiring infants are exposed to: raw acoustic data. A GAN model trained on an allophonic distribution is successful in learning to generate acoustic output from random noise. The generated acoustic outputs include evidence that the Generator network learns the conditioned distribution of VOT duration. Additionally, the model actually outputs unique acoustic data that can be investigated with acoustic phonetic methodology, allowing a direct comparison between human speech data and GAN's generated output.

Modeling human behavior with neural network has seen a rapid expansion in recent years. An increasing number of works argues that neural network modeling resemble the actual neural processing of vision (Peterson et al., 2018; Bashivan et al., 2019). This paper aims to be a first step in expanding this discussion to acoustic speech signal.

2 Materials

2.1 Model

Generative Adversarial Networks, proposed by Goodfellow et al. (2014), have seen a rapid expansion in a variety of tasks, including but not limited to computer vision and image generation (Radford et al., 2015). 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 (e.g. image pixels) from random noise, while the Discriminator is trained on distinguishing real data from the 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 as image pixels) such that the Discriminator is inaccurate in distinguishing the generated from real data. Goodfellow et al. (2014) summarizes the architecture (repeated here in Equation 4), where V is value function that the Generator maximizes and Discriminator minimizes, G is Generator, D is Discriminator, x is data from $P_{\text{data}}(x)$, z is noise input variable from prior P_z.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim P_{\text{data}}(x)}[\log(D(x)] + \mathbb{E}_{z\sim P_z(z)}[1-\log(D(G(z)))] \tag{4}$$

Applying the GAN architecture on a time-series data such as continuous speech stream faces several challenges. Recently, Donahue et al. (2019) proposed an implementation of a Deep Convolutional Generative Adversarial Network proposed by Radford et al. (2015) for audio data (WaveGAN). The model takes one-second long raw audio files as inputs, sampled at 16 kHz with 16-bit quantization. The audio files are converted into a vector and fed to the Discriminator network as real data. Instead of the two-dimensional 5×5 filters, WaveGAN model uses one-dimensional 1×25 filters and larger upsampling. The main architecture is preserved as in DCGAN, except that an additional layer is introduced in order to generate longer samples. The Generator network takes as input z, a vector of one hundred uniformly distributed variables ($Uniform(-1, 1)$ 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 initial GAN design as proposed by Goodfellow et al. (2014) trained the Discriminator network on distinguishing real from generated data. Training such models, however, faced substantial challenges (Donahue et al., 2019). Donahue et al. (2019) implements WGAN-GP strategy (Arjovsky et al., 2017; Gulrajani et al., 2017), which means that the Discriminator is trained "as a function that assists in computing the Wasserstein distance" (Donahue et al., 2019). 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). For exact dimensions of each layer and other details of the model, see Donahue et al. (2019).

2.2 Training data

The model was trained on allophonic distribution of voiceless stops in English. As already mentioned in Section 1, voiceless stops /p, t, k/ surface as aspirated [p^h, t^h, k^h] in English in word-initial position when immediately followed by a stressed vowel. If an alveolar sibilant [s] precedes the stop, however, the aspiration is blocked and the stop surfaces as unaspirated [p, t, k]. A minimal pair illustrating this allophonic distribution is [ˈp^hɪt] 'pit' vs. [ˈspɪt] 'spit'. The most prominent phonetic correlate of this allophonic distribution is the difference in VOT duration (Abramson and Whalen, 2017) between the aspirated and unaspirated voiceless stops.

Model was trained on data from the TIMIT database (S Garofolo et al., 1993).[2] The corpus was chosen because it is one of the largest currently available hand-annotated speech corpora. The database includes 6300 sentences, 10 sentences per 630 speakers from 8 major dialectal areas in the

[2] Donahue et al. (2019) train the model on SC09 and TIMIT databases, but the results are not useful for modeling phonological learning, because the model is trained on continuous speech stream and the generated sample fail to produce analyzable results for phonological purposes.

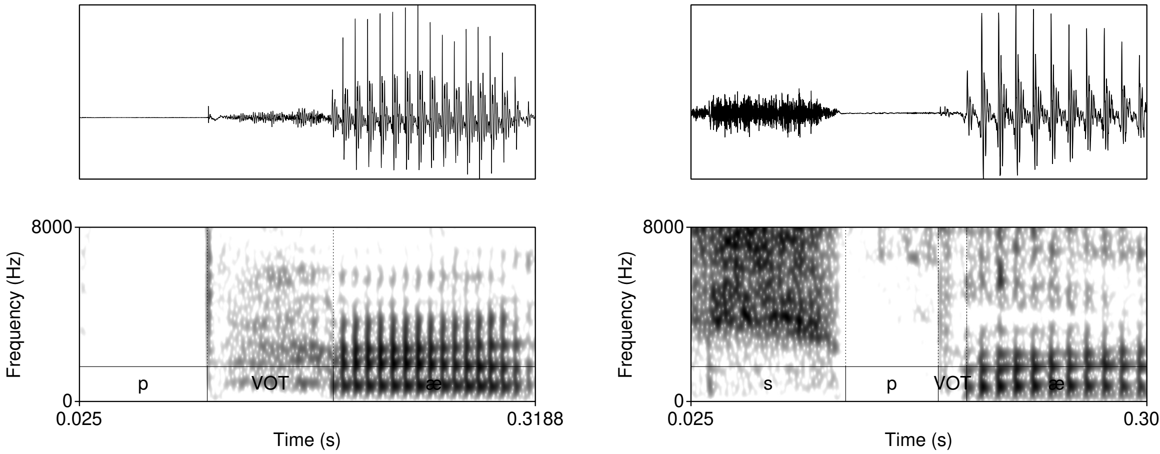

Figure 1: Waveforms and spectrograms ($0 - 8000$ Hz) of [p^{h}æ] (left) and [spæ] (right) illustrating typical training data with annotations from TIMIT. Only the raw audio data (in .wav format) were used in training. The annotation illustrates a substantially longer duration of VOT in word-initial stops when no [s] precedes.

US (S Garofolo et al., 1993). The training data consist of 16-bit .wav files with 16 kHz sampling rate of word initial sequences of voiceless stops /p, t, k/ (= T) that were followed by a vowel (#TV) and word initial sequences of /s/ + /p, t, k/, followed by a vowel (#sTV). The training data includes 4,930 sequences with the structure #TV and 533 sequences with the structure #sTV (5,463 total). Figure 1 illustrates typical training data: raw audio files with speech data, but limited to two types of sequences, #TV and #sTV. Figure 1 also illustrates that the duration of VOT depends on a condition that is not immediately adjacent in phonetic terms: absence/presence of [s] is interrupted from the VOT duration by a period of closure in the training data. It is not immediately obvious whether a Generative Adversarial Neural network should be capable of capturing such non-local conditions in duration.

Both stressed and unstressed vowels are included in the training data. Including both stressed and unstressed vowels is desirable, as this condition crucially complicates learning and makes the task for the neural network more challenging. Aspiration is less prominent in word-initial stops not followed by a stressed vowel. This means that in the condition #TV, the stop will be either fully aspirated (if followed by a stressed vowel) or not fully aspirated (if followed by a unstressed vowel). In the #sTV condition, the stop is never aspirated. Learning of two conditions is more complex if the dependent variable in one condition can range across the variable in the other condition.

To confirm the presence of this durational distribution in the training data, VOT duration was measured across the two conditions. Hand annotations in the TIMIT database were used for measuring VOT durations. VOT is measured from the release of the stop to the onset of the following vowel. Slices for which no VOT duration exists in TIMIT (only closure duration that includes the VOT) were excluded from this analysis: altogether 47 sequences were thus excluded. While the TIMIT database is occasionally misaligned, the errors are not substantial to crucially affect the outcomes. Table 3 and Figure 2 summarize raw VOT durations across three places of articulation. Speaker identity is not included in the model, because for the purpose of training a GAN network, speaker information is irrelevant.

To test significance of the presence of [s] as a predictor of VOT duration, data were fit to a linear

Structure	Place	VOT	SD	Lowest	Highest
#TV	p	49.6	18.0	7.3	115.5
	t	55.2	20.7	9.8	130.0
	k	67.5	19.5	12.5	153.1
#sTV	p	19.4	7.1	9.4	49.2
	t	25.6	7.9	10.6	65.0
	k	30.1	8.6	14.4	55.0

Table 3: Raw VOT durations in ms for the training data with SD and Range.

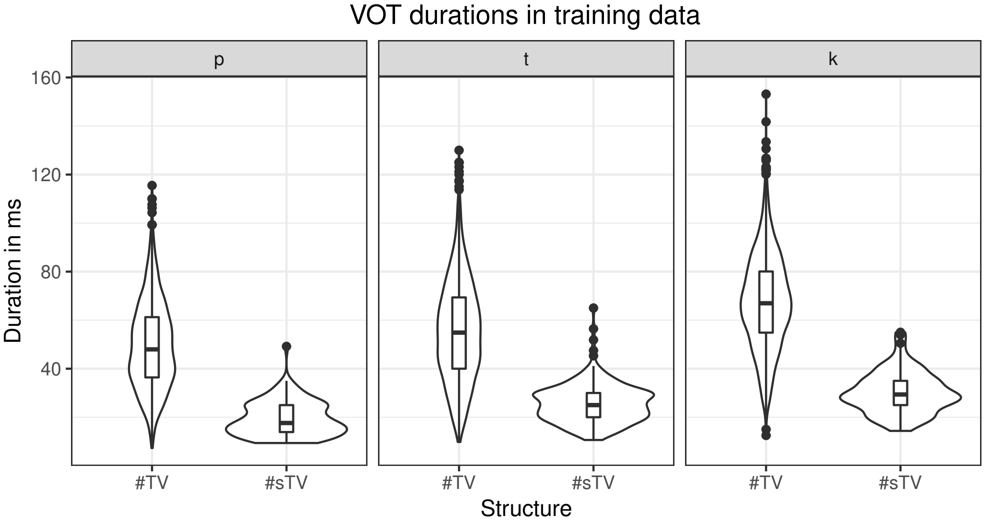

Figure 2: Violin plots with box-plots of durations in ms of VOT in the training data based on two conditions: when word-initial TV sequence is not preceded by [s] (#sTV) and when it is preceded by [s] (#sTV) accross the three places of articulation: [p], [t], [k].

model with two predictors: Structure (presence vs. absence of [s]) and Place of articulation of the target stop (with three levels — [p], [t], [k]) and their interaction. Structure was treatment-coded (with absence of [s] as the reference level), while place of articulation of the stop was sum-coded (with [k] as reference). The interaction term is significant ($F(2) = 6.97, p < 0.001$), which is why it is kept in the final model. The model shows that at the mean of the Place of articulation as a predictor, VOT is approximately 32.4 ms shorter if T is preceded by [s]. The 95% confidence intervals for this difference are [−34.3 ms, −30.6 ms]. Figure 3 illustrates the significant difference and its magnitude between the two conditions across the three places of articulation. The significant interaction #sTV:[t] is not informative and irrelevant for our purposes.

The training data is not a completely naturalistic: only #TV and #sTV sequences are sliced from continues speech data. This, however, has a desirable effect. The primary purpose of this paper is to test whether a GAN model can learn an allophonic distribution from data that consists of raw acoustic inputs. If the whole lexicon were included in the training data, the distribution of VOT duration could be conditioned on some other distribution, not the one this paper is predominately interested in: presence or absence of [s]. It is thus less likely that the distribution of VOT duration across the main condition of interest, presence of [s], is influenced by some other unwanted factor precisely because of the balanced design of the training data. The only condition that can influence the outcomes is the distribution of vowels across the two conditions. Figure 4, however, shows that

	β	SE	t-value	Pr(>|t|)
(Intercept)	57.4	0.28	203.37	0.0000
#TV vs. #sTV	-32.4	0.95	-34.16	0.0000
[p] vs. mean	-7.8	0.44	-17.64	0.0000
[t] vs. mean	-2.2	0.38	-5.79	0.0000
#sTV:[p]	2.2	1.44	1.49	0.1357
#sTV:[t]	2.8	1.20	2.30	0.0213

Table 4: Linear model

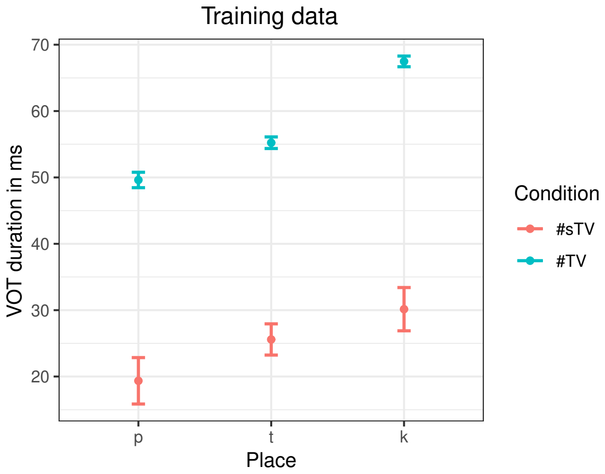

Figure 3: Distribution of VOT durations as estimated from a linear model with

vowels are relatively equally distributed across the two conditions, which means that vowel identity likely does not influence the outcomes substantially. Finally, speech rate is not controlled for in the present experiment. To control for speech rate, VOT duration would have to be modeled as a proportion of the following vowel duration. Several confounds that are not easy to address would be introduced, the main of which is that vowel identification is not unproblematic for generated inputs with fewer training steps 3.2. Because the primary interest of the experiment is the difference in VOT durations between two groups (presence and absence of [s]) and substantial differences in speech rate between the two groups are not expected, we do not anticipate the results to be substantially influenced by speech rate.

Also, the training is is performed on already sliced #TV and #sTV sequences, padded with silences, rather than on continuous speech data. This also should not pose a significant problem to the unsupervised learning mode in this paper. One can imagine a separate model that learns to distinguish silences from acoustic speech signal and performs learning on speech signal only. The current model is skipping this step and feeding sliced acoustic speech signal as unsupervised training data.

3 Experiment

3.1 Training and generation

The model was trained on a single NVIDIA K80 GPU. The network was trained at an approximate pace of 40 steps per 300 s. The purpose of this paper is to model phonetic and phonological

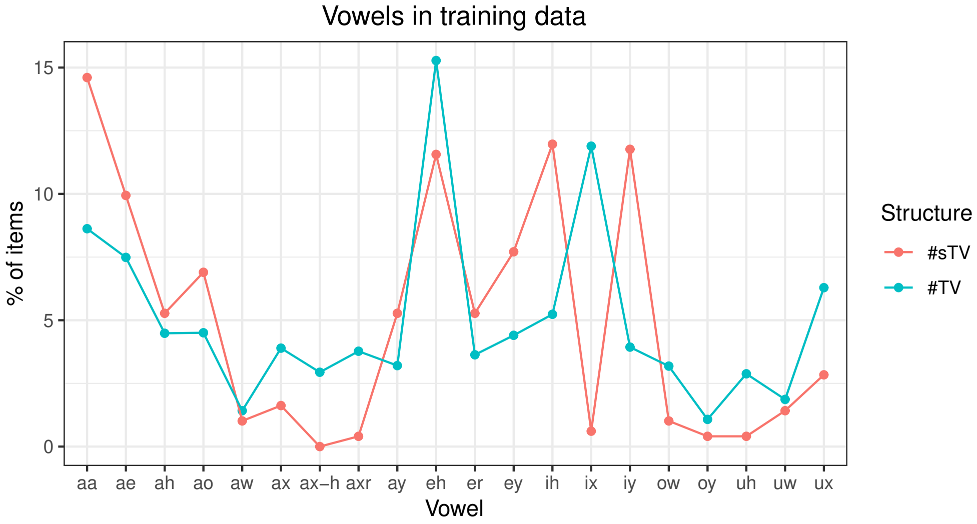

Figure 4: Distribution of training items according to vowel identity as described in TIMIT in ARPABET, where aa = ɑ, ae = æ, ah = ʌ, ao = ɔ, aw = aʊ, ax = ə, ax-h = ə̥, axr = ɚ, ay = aɪ, eh = ɛ, er = ɝ, ey = eɪ, ih = ɪ, ix = ɨ, iy = i, ow = oʊ, oy = oɪ, uh = ʊ, uw = u, ux = ʉ in International Phonetic Alphabet.

learning. For this reason, the Generator network was not fully trained until convergence: the data was generated and examined as the Generator network was progressively being trained.

3.2 Model 1: 1,474 steps

In the first test of the model, the network was trained with 1474 steps (approximately 86 epochs).[3] The Generator network generated 950 samples (.wav files). Every generated output was listened to and spectral properties were manually observed by the author. At this point, the model is performing poorly, which is why only qualitative analysis of the generated samples is possible. Nevertheless, some significant observation emerge even in this initial model.

Most of the generated samples already have a clear vocalic element with more or less pronounced formant structure and a non-vocalic element — VOT after the release of closure. Figures 5 illustrates a typical output with the structure #TV. The spectrograms show both vocalic structure and frication noise from aspiration. VOT duration is substantial. The Generator also generates sequences with the structure #sTV, illustrated in Figure 6. The peculiarity about the #sTV sequences at this point is that the sibilant part seems substantially shorter (with a narrow band of [s]-like frequency distribution) and the closure features relatively high amount of noise. This limited sample already suggest that the Generator might be learning the conditional VOT distribution as outputs with [s] feature no obvious VOT duration (although bursts are not clearly visible either).

At this point, the Generator network also generate samples that substantially violate distributions in the training data. One such output includes three consecutive sibilants [sss]; another includes two or three consecutive vocalic elements divided by periods of reduced noise (Figure 7). Occasionally, the order of segments is violated. The left spectrogram in Figure 6 shows that a short vocalic element surfaces between [s] and the closure. The left spectrogram in Figure 7 shows that a period of silence (marked with an arrow) intervenes during the vowel V.

[3] Metagraph with 6,759 steps was used for generation.

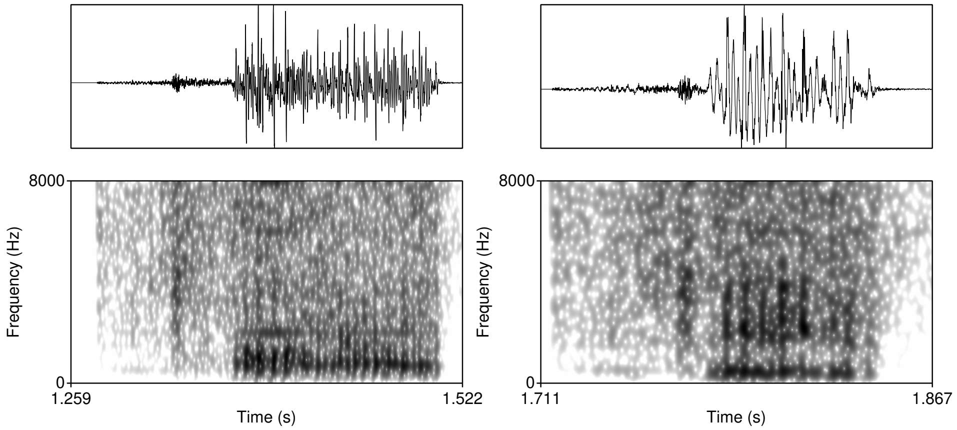

Figure 5: Waveforms and spectrograms (0–8,000 Hz) of a typical generated samples of #TV sequences from a Generator trained after 1474 steps.

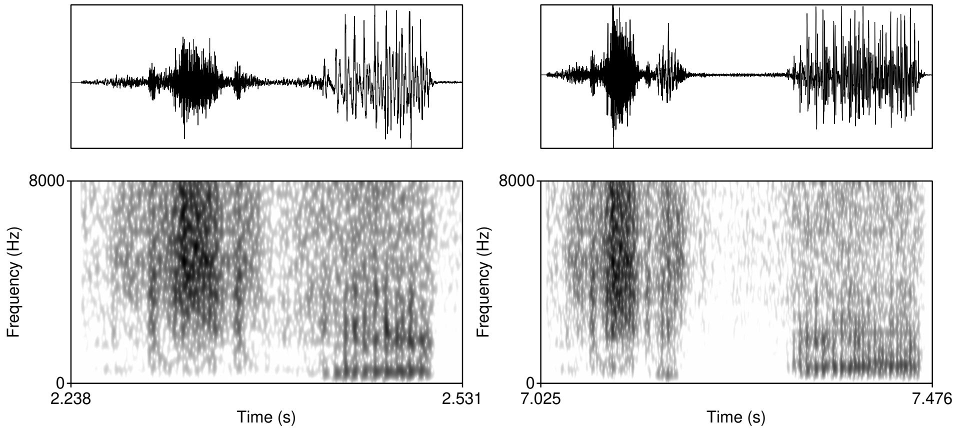

Figure 6: Waveforms and spectrograms (0–8,000 Hz) of a typical generated samples of #sTV sequences from a Generator trained after 1474 steps.

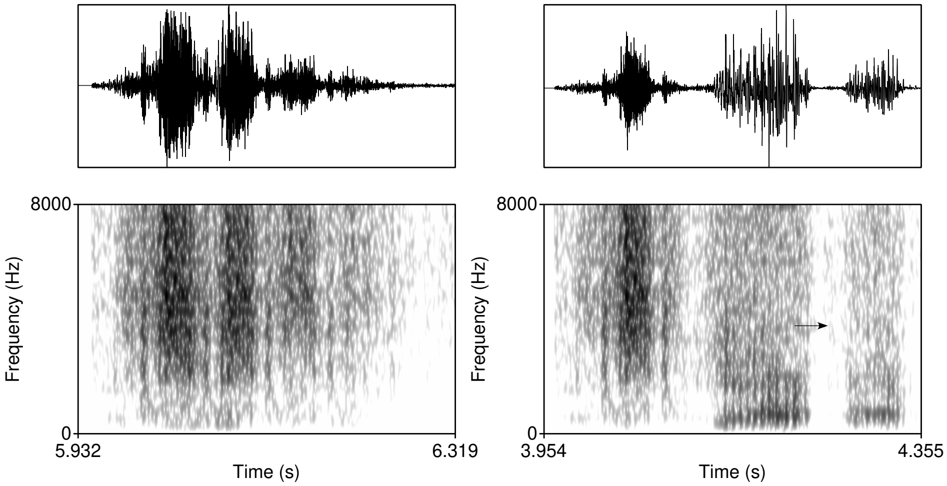

Figure 7: Waveforms and spectrograms (0–8,000 Hz) of generated samples that violate training data distributions from a Generator trained after 1474 steps. The left spectrogram shows a sequence of three [s] divided by periods of reduced frication noise. The right spectrogram illustrates silence (marked with an arrow) during the vocalic element.

3.3 Model 2: 12,255 steps

The Generator network after 12,225 steps ($\sim$ 716 epochs) generates speech signal that appears substantially closer to actual acoustic data compared to Model 1. Figure 8 illustrates a typical generated sample of #TV (left) and #sTV (right) structures. VOT durations are substantially different.

To test whether the Generator learns the conditional distribution of VOT duration, the Generated samples were annotated for VOT duration. VOT duratuon was measured from the release of closure to the onset of periodic vibration with clear formant structure. 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. Only samples with structure that resembles real acoustic outputs were annotated. Figure 9 shows raw distribution of VOT durations in the generated samples that closely resembles the distribution in the training data (Figure 2).

To test significance of the observed distribution, the generated data were fit to a linear model with only one predictor: absence of [s] (Structure). Place of articulation or following vowel were not added in the model, because it is often difficult to recover place of articulation or vowel quality of generated samples. Structure 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, $t = -7.29, p < 0.0001$). Figure 10 illustrates estimates of VOT duration across the two conditions with 95% confidence intervals.

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 fails to learn the distribution categorically at 12,255 steps. The three longest VOT durations in the #sTV 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; see Table 3 and Figure 2). Figure 11 shows one such case. It is clear that the generator fails to reproduce

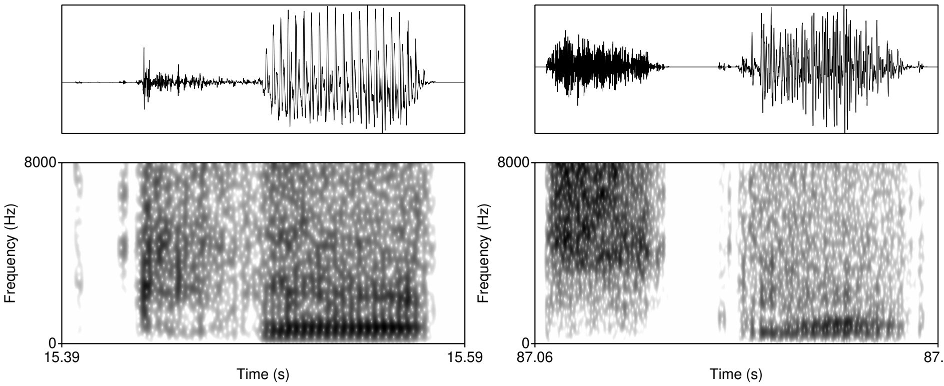

Figure 8: Waveforms and spectrograms (0–8,000 Hz) of a typical generated samples of #TV (left) and #sTV (right) sequences from a Generator trained after 12,255 steps.

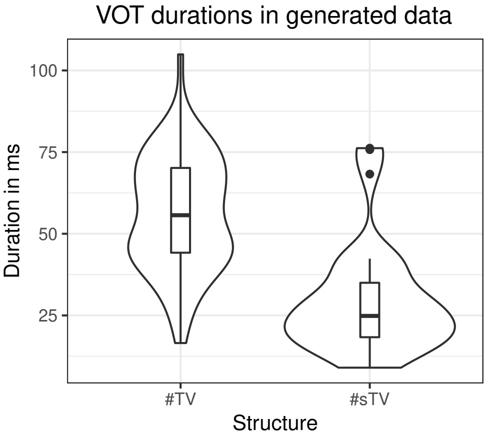

Figure 9: Violin plots with box-plots of durations in ms of VOT in the generated data based on two conditions: when word-initial TV sequence is not preceded by [s] (#sTV) and when it is preceded by [s] (#sTV).

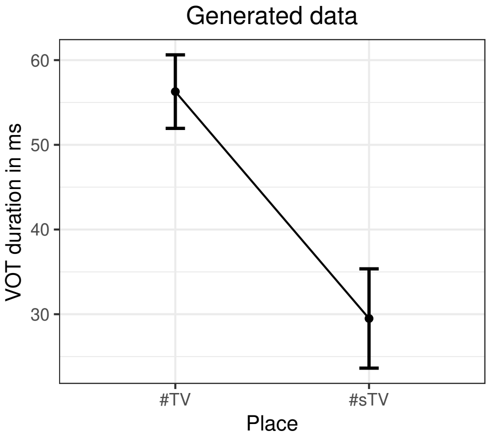

Figure 10: Estimates of VOT duration with 95% confidence intervals across two conditions, #TV and #sTV in the generated data for a model trained after 12,255 steps.

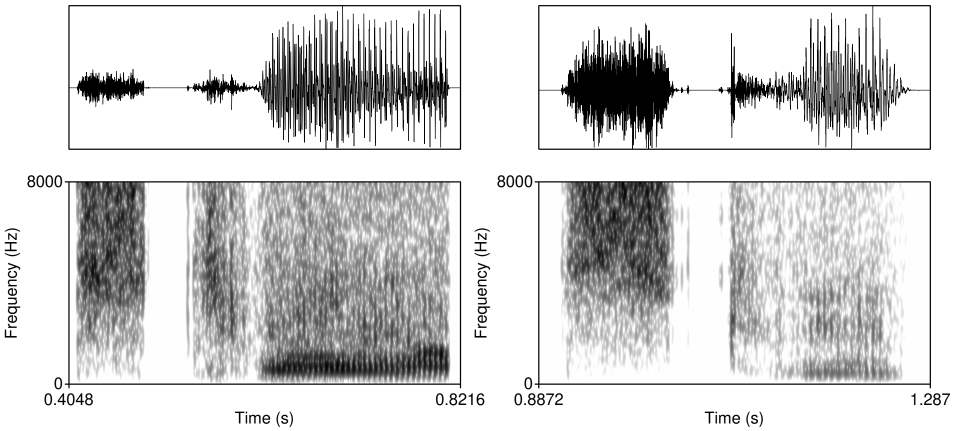

Figure 11: Waveforms and spectrograms (0–8000 Hz) of two generated outputs of #sTV sequences in which the stop has longer VOT than any VOT in #sTV condition in the training data.

the conditioned durational distribution from the training data in this particular case.

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 by the author and none of them contain a #sV sequence without a period of stop and VOT. Homorganic sequences of [s] followed by an alveolar stop [t] (#stV) are occasionally acoustically similar to the sequence without the stop (#sV) because frication noise from [s] carries onto the homorganic alveolar closure which can be very short. However, there is a clear fall and a second rise of noise amplitude after the release of the stop in #stV sequences. Figure 12 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.

Measuring overfitting is a substantial problem for Generative Adversarial Networks with no consensus on the most appropriate quantitative approach to the problem. The danger with overfitting in a GAN is that the Generator network would learn to fully replicate the input. There are several reasons to believe that our Generator does not overfit. Perhaps the best evidence against overfitting is the fact that the Generator network outputs samples that substantially violate output distributions (Figures 11 and 12).

4 Discussion

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. These results suggest that the phonetic and phonological learning (of at

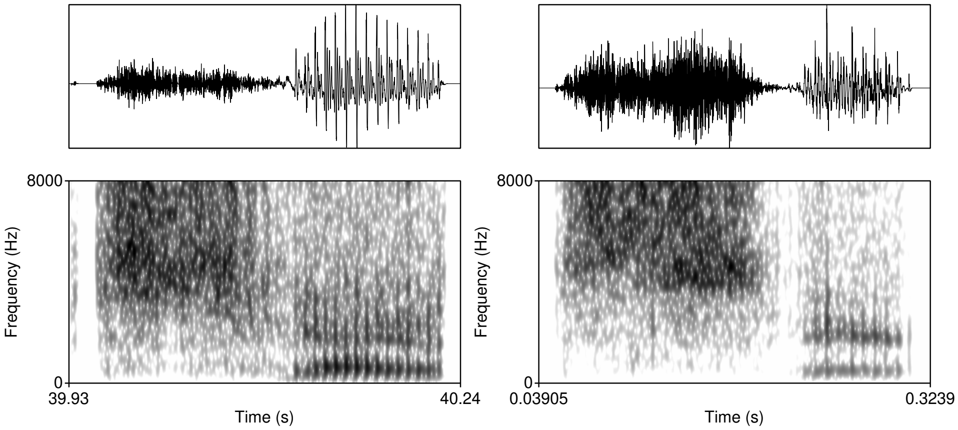

Figure 12: Waveforms and spectrograms (0–8000 Hz) of two generated outputs of the shape #sV sequences for which no evidence was present in the training data. The sample on the left was generated after 16,715 steps.

least non-contrastive allophonic disgributions) can be modeled with Generative Adversarial Neural networks. The advantage of this proposal is that learning is not modeled already at the symbolic level, but from raw phonetic inputs. GAN's architecture resembles the production-perception loop: production is modeled with the Generator network, perception with the Discriminator network.

To be sure, this paper is just a first step in arguing that phonological grammar should be modeled on raw speech data rather than on symbolic representation and that the generative adversarial approach can offer crucial insights for this task. The current model does not operate with phonemic representation yet: the Generator network learns a distributional allophonic variation. The Generator does, however, show evidence for categorical learning: some of the generated samples that violate input data resemble human linguistic behavior and suggest that the network learns [s] and V to be discrete units that can be combined together without an intervening stop. With further training, such irregular outputs should be eliminated.

Also, the model does not contain any articulatory information. The Generator network is not limited by confines of human articulatory apparatus. The outputs of the Generator network can thus be interpreted as articulatory targets, which would be sent through human articulators to produce actual speech. In a more speculative interpretation, the latent variables can be interpreted as neural signal that the Generator learns to output into meaningful targets for execution of articulatory process (for a recent work on brain signals of speech stream, see).

5 Conclusions and future directions

This paper presents a novel model of unsupervised phonetic and phonological learning that is based on the Generative Adversarial Neural network architecture (Goodfellow et al. 2014; Radford et al. 2015, implemented for audio inputs in Donahue et al. 2019). One of the main advantages of the proposed architecture is that learning is modeled in an unsupervised mode from raw acoustic data. The model does not require a level of symbolic representation, as is the case for the majority of current proposals.

Applying this method on novel input data with evidence for various phonetic and phonological processes should yield a better understanding of the nature of phonetic and phonological learning, phonological computation, and grammar in general. Phonetic and phonological language acquisition can be directly compared to outputs of the Generator to identify similarities and divergences. Speech errors are another aspect that can be directly compared to the outputs of the Generator network for commonalities and divergences.

Modeling of phonological learning with GANs should provide information about which aspects of phonological grammar and learning can be modeled with approaches that contain no language-specific devices. Comparing human phonetic and phonological behavior with outputs of the Generative Adversarial Networks should provide direct answers to such questions.

Acknowledgements

This research was funded by a grant to new faculty at the University of Washington. I would like to thank Sameer Arshad for slicing data from the TIMIT database. All mistakes are my own.

References

Abramson, A. S., Whalen, D., 2017. Voice onset time (vot) at 50: Theoretical and practical issues in measuring voicing distinctions. Journal of Phonetics 63, 75 – 86.
URL `http://www.sciencedirect.com/science/article/pii/S0095447016301048`

Albright, A., Hayes, B., 2011. Learning and Learnability in Phonology. Wiley, Ch. 20, pp. 661–690.
URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/9781444343069.ch20`

Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein generative adversarial networks. In: International Conference on Machine Learning. pp. 214–223.

Avcu, E., Shibata, C., Heinz, J., 2017. Subregular complexity and deep learning. In: Proceedings of the Conference on Logic and Machine Learning in Natural Language (LaML).

Bashivan, P., Kar, K., DiCarlo, J. J., 2019. Neural population control via deep image synthesis. Science 364 (6439).
URL `https://science.sciencemag.org/content/364/6439/eaav9436`

Chomsky, N., Halle, M., 1968. The Sound Pattern of English. Harper & Row, New York.

Dell, G. S., Juliano, C., Govindjee, A., 1993. Structure and content in language production: A theory of frame constraints in phonological speech errors. Cognitive Science 17 (2), 149 – 195.
URL `http://www.sciencedirect.com/science/article/pii/0364021393900106`

Donahue, C., McAuley, J., Puckette, M., 2019. Adversarial audio synthesis. In: ICLR.

Goldwater, S., Johnson, M., 2003. Learning OT constraint rankings using a maximum entropy model. In: Spenader, J., Eriksson, A., Dahl, O. (Eds.), Proceedings of the Workshop on Variation within Optimality Theory. Stockholm University, Stockholm, pp. 111–20.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., Weinberger, K. Q. (Eds.), Advances in Neural Information Processing Systems 27. Curran Associates, Inc., pp. 2672–2680.
URL `http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A. C., 2017. Improved training of wasserstein gans. In: Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 30. Curran Associates, Inc., pp. 5767–5777.
URL `http://papers.nips.cc/paper/7159-improved-training-of-wasserstein-gans.pdf`

Hayes, B., 1999. Phonetically-driven phonology: The role of Optimality Theory and inductive grounding. In: Darnell, M., Moravscik, E. (Eds.), Functionalism and Formalism in Linguistics, Volume I: General Papers. John Benjamins, Amsterdam, pp. 243–285.

Hayes, B., White, J., 2013. Phonological naturalness and phonotactic learning. Linguistic Inquiry 44 (1), 45–75.

Hayes, B., Wilson, C., 2008. A maximum entropy model of phonotactics and phonotactic learning. Linguistic Inquiry 39 (3), 379–440.

Heinz, J., 2011. Computational phonology – part ii: Grammars, learning, and the future. Language and Linguistics Compass 5 (4), 153–168.
URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1749-818X.2011.00268.x`

Iverson, G. K., Salmons, J. C., 1995. Aspiration and laryngeal representation in germanic. Phonology 12 (3), 369–396.
URL `http://www.jstor.org/stable/4420084`

Jarosz, G., 2019. Computational modeling of phonological learning. Annual Review of Linguistics 5 (1), 67–90.
URL `https://doi.org/10.1146/annurev-linguistics-011718-011832`

Kawamoto, A. H., Liu, Q., Kello, C. T., 2015. The segment as the minimal planning unit in speech production and reading aloud: evidence and implications. Frontiers in Psychology 6, 1457.
URL `https://www.frontiersin.org/article/10.3389/fpsyg.2015.01457`

Kim, C.-W., 1970. A theory of aspiration. Phonetica 21, 107–116.

Legendre, G., Miyata, Y., Smolensky, P., 1990. Harmonic grammar: A formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations. University of Colorado, Boulder. ICS Technical Report #90-5.

Legendre, G., Sorace, A., Smolensky, P., 2006. The Optimality Theory—Harmonic Grammar connection. In: Smolensky, P., Legendre, G. (Eds.), The Harmonic Mind: From Neural Computation to Optimality-Theoretic Grammar. MIT Press, Cambridge, MA, pp. 339–402.

Mahalunkar, A., Kelleher, J. D., 2018. Using regular languages to explore the representational capacity of recurrent neural architectures. In: Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L., Maglogiannis, I. (Eds.), Artificial Neural Networks and Machine Learning – ICANN 2018. Springer International Publishing, Cham, pp. 189–198.

Moreton, E., Pater, J., Pertsova, K., 2017. Phonological concept learning. Cognitive Science 41 (1), 4–69.
URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/cogs.12319`

Pater, J., 2009. Weighted constraints in generative linguistics. Cognitive Science 33, 999–1035.

Pater, J., 2019. Generative linguistics and neural networks at 60: Foundation, friction, and fusion. Language.

Peterson, J. C., Abbott, J. T., Griffiths, T. L., 2018. Evaluating (and improving) the correspondence between deep neural networks and human representations. Cognitive Science 42 (8), 2648–2669. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/cogs.12670

Prickett, B., Traylor, A., Pater, J., 2019. Learning reduplication with a variable-free neural network, ms., University of Massachusetts, Amherst. http://works.bepress.com/joe_pater/38/ (accessed 23 May 2019).

Prince, A., Smolensky, P., 1993/2004. Optimality Theory: Constraint Interaction in Generative Grammar. Blackwell, Malden, MA, first published in 1993, Tech. Rep. 2, Rutgers University Center for Cognitive Science.

Radford, A., Metz, L., Chintala, S., 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.

Rawski, J., Heinz, J., 2019. No free lunch in linguistics or machine learning: Response to pater. Language.

S Garofolo, J., Lamel, L., M Fisher, W., Fiscus, J., S. Pallett, D., L. Dahlgren, N., Zue, V., 11 1993. Timit acoustic-phonetic continuous speech corpus. Linguistic Data Consortium.

Schatz, T., Feldman, N., Goldwater, S., Cao, X. N., Dupoux, E., May 2019. Early phonetic learning without phonetic categories – insights from machine learning.
URL psyarxiv.com/fc4wh

Weber, N., Shekhar, L., Balasubramanian, N., 2018. The fine line between linguistic generalization and failure in Seq2Seq-attention models. In: Proceedings of the Workshop on Generalization in the Age of Deep Learning. Association for Computational Linguistics, New Orleans, Louisiana, pp. 24–27.
URL https://www.aclweb.org/anthology/W18-1004

White, J., 2014. Evidence for a learning bias against saltatory phonological alternations. Cognition 130 (1), 96 – 115.
URL http://www.sciencedirect.com/science/article/pii/S0010027713001923

White, J., 2017. Accounting for the learnability of saltation in phonological theory: A maximum entropy model with a P-map bias. Language 93 (1), 1–36.

Wilson, C., 2006. Learning phonology with substantive bias: An experimental and computational study of velar palatalization. Cognitive Science 30, 945–982.