A SYSTEMATIC EVALUATION OF COMPUTATIONAL MODELS OF PHONOTACTICS

By

Isaac Sarver

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Linguistics – Master of Arts

2020

ABSTRACT

A SYSTEMATIC EVALUATION OF COMPUTATIONAL MODELS OF PHONOTACTICS

By

Isaac Sarver

In this thesis, recent computational models of phonotactics are discussed and evaluated and two new models are implemented. Prior phonotactic modeling, motivated by gradient acceptability judgments in nonce word judgment tasks (Albright 2009), claim that phonotactic grammaticality is gradient, and these models are evaluated by their ability to judge nonce words with scores that correlate with human acceptability judgments. Gorman (2013) argues that these gradient models do not account for the facts sufficiently and claims phonotactic grammaticality is categorical. In this thesis, the account of Gorman (2013) is implemented as well as a prominent gradient model from Hayes and Wilson (2008) and compared with the performance of two machine learning models (a support vector machine and a recurrent neural network), with all models trained on a corpus of English onsets. Results in this thesis show that the computational models are unable to correlate with human judgment data from Scholes (1966) as well as a categorical prediction of acceptability based on whether a sequence is attested in the lexicon or not, and that these models rely on assumptions which when challenged show that the models do not convincingly capture the gradience of the human judgment data used for evaluation.

This thesis is dedicated to my siblings.

ACKNOWLEDGEMENTS

First and foremost I would like to thank the linguistics faculty for their support and guidance throughout my time at MSU. This includes Hannah Forsythe, who taught my Introduction to Language course in Spring 2015 and first introduced the field of linguistics to me, as well as Marcin Morzycki, who taught my second linguistics class I took and encouraged me to add a linguistics minor in my undergrad, and who also advised me throughout my semantics research my first year of grad school. I am hugely indebted as well to Suzanne Wagner, Yen-Hwei Lin, Alan Munn, and Cristina Schmitt for all that I've learned in the program during classes, colloquiums, and personal discussion. Thank you also to Kristen Johnson in the CSE department for NLP advice and instruction. And special thanks to my advisor, Karthik Durvasula, who persuaded me to apply to the MA program, taught four of my classes, and gave me support and advice throughout the whole process.

Thank you also to my fellow students. I'm so grateful for the friendships and discussions of this time in classes, the grad office, and colloquium dinners. Thank you to those who listened to my ideas and presented their own in Awkward Time, the Phono group, and informal study sessions. And thank you as well to my advisors and colleagues in the Center for Language Teaching Advancement who have given me so many professional opportunities this year.

Lastly, there is no way I would have made it through the last few years without my friends and family. Thank you to Megan Wixom and Daniela Diaz for listening to all of my practice presentations, thank you to Suzanna Feldkamp for keeping me accountable through our late-night writing sessions, and thank you to Abdullah Karaaslanlı and Thanaphong Phongpreecha for being daily listening ears for everything I had on my mind and for helping me through learning Python and machine learning. Thank you to my siblings for giving me the ability to laugh on any day no matter the circumstances, and to my parents who have supported me and fostered my curiosity about the world from the beginning. Thank you all.

TABLE OF CONTENTS

LIST OF TABLES vi

LIST OF FIGURES vii

1 INTRODUCTION 1

2 RELATED WORKS 5
- 2.1 Origins of phonotactics 5
- 2.2 Nature of constraints and output 7
- 2.3 Dealing with Gradient Acceptability: Recent Models 9
- 2.4 Frequency structure in the data 11

3 METHODS 14
- 3.1 Phonotactic models for this study 14
- 3.2 Data Preparation 15
- 3.3 SVM model 17
- 3.4 RNN model 20
- 3.5 Maximum Entropy grammar 22

4 RESULTS 26
- 4.1 Gross Phonotactic Violation 26
- 4.2 SVM results 27
- 4.3 RNN results 28
- 4.4 Changing frequency structures in the data for RNN training 31
- 4.5 Changing frequency structures in the data for MaxEnt training 32

5 DISCUSSION 35
- 5.1 Model Comparisons 35
- 5.2 Training data structure 36
- 5.3 Experiment data used for evaluation 37

6 CONCLUSION 38

BIBLIOGRAPHY 41

LIST OF TABLES

Table 2.1: Examples of the nonce words used in Scholes' experiment, along with number of positive responses. 6

Table 3.1: Onsets used for training 16

Table 3.2: Toy example of different frequency structures in training data. 17

Table 3.3: Maxent Grammar (note: '.' represents multiplication here) 22

Table 4.1: Type frequency model results on withheld test set 32

Table 4.2: Equalized Frequency Model results on withheld test set 32

Table 4.3: Type frequency model results predicting Scholes data 32

Table 4.4: Equalized frequency model results predicting Scholes data 32

Table 4.5: Correlations of the Phonotactic Learner scores with the Scholes data . . . 34

LIST OF FIGURES

Figure 2.1: Mean ratings of subjects asked how representative the stimuli were of the categories "even" and "odd" respectively, where "1" indicates most representative (Gorman 2013; Armstrong, L. Gleitman, and H. Gleitman 1983). 8

Figure 3.1: Illustration of an SVM in 2 dimensions 18

Figure 3.2: Sketch of the input nodes, hidden layer, and output of a Recurrent Neural Network. 20

Figure 3.3: Graph representation of the sigmoid function, with x = input and y = output. 22

Figure 4.1: Distribution of normalized ratings in the Scholes experiment for attested and unattested onsets. 27

Figure 4.2: The accuracy of the SVM model's classifications of the test data based on the type of embedding. 28

Figure 4.3: SVM results on withheld test set 29

Figure 4.4: SVM predictions for Scholes data, with ground truth set to gross status . 29

Figure 4.5: Loss values during RNN training, showing how wrong the model is for each iteration. 30

Figure 4.6: Confusion matrix showing the model's accuracy for guessing each class, and its error. 30

Figure 4.7: Scatter plot showing the model's % confidence that an onset is grammatical (value=1), compared to the percentage of "yes" responses in the Scholes experiment. 31

Figure 4.8: Equalized Frequency, Tuned 34

Figure 4.9: Equalized Frequency, Untuned 34

Figure 4.10: Type Frequency, Tuned 34

Figure 4.11: Type Frequency, Untuned 34

CHAPTER 1

INTRODUCTION

Phonotactic theory is concerned with understanding the knowledge that a speaker has about possible and impossible phonological sequences in their native language. This knowledge allows for speaker judgments of novel words, where, upon hearing a novel sequence of sounds, a speaker can determine if the new word is an acceptable sequence that could reasonably be a member of their lexicon or not. For example, when speakers are presented with the non-English words <blick> [blɪk] and <bnick> [bnɪk], speakers can easily assess <blick> to be well-formed and <bnick> to be ill-formed.[1] Note, however, that the reason for their exclusion from the lexicon is categorically different. The former is acceptable, yet happens to not occur in the lexicon of most speakers; an accidental gap in the lexicon rather than a systemic one. The latter, on the other hand, violates some structural requirement, and is judged to be an impossible construction in the language (Halle 1962; Chomsky and Halle 1965).

In recent work, phonotactic judgments have been argued to be based on gradient knowledge (Albright and Hayes 2003; Albright 2009). For example, [wɪs] and [ploʊmf] can both be judged as acceptable, but [wɪs] is consistently judged as 'more acceptable' than [ploʊmf]. This can be considered the more prominent view, often adopted in modern research on phonotactic knowledge. Generally, this view equates gradient acceptability with gradient grammaticality[2].

There has also been some recent work concerned with the potential of modeling phonotactic knowledge with machine learning and deep learning techniques (Mayer and Nelson 2019; Mirea and Bicknell 2019). This follows the current enthusiasm for the ways deep learning

[1] <...> denotes orthographic representations, [...] denotes surface representations, and /.../ denotes underlying representations.

[2] Although, as many have argued, gradient acceptability does not necessitate gradient grammaticality(Gorman 2013; Chomsky 1965; Schütze 2011)

and more powerful computational methods might provide windows into linguistic theory and knowledge (Pater 2019). Phonotactic knowledge is particularly well-suited to this approach, because any phonotactic model can be based on a small number of features, whether segmental or featural, and only need produce a simple output: in a categorical system, valid or invalid; in a gradient system, some probability of a sequence's acceptability.

In this thesis, I show that recent computational models in the phonotactics literature, as well as my own models, can learn phonotactic generalizations from corpus data equally well without any information in the data regarding the frequency of a sequence in the lexicon. This suggests that gradient phonotactic acceptability judgments are not the result of varying sequence frequencies in the lexicon; that lexical frequency ratios of phonological sequences are perhaps not relevant to modeling speakers' phonotactic judgments.

In a dissertation that looked specifically at different types of models that can account for acceptability judgments, Gorman (2013) argued that a categorical baseline can outperform gradient models such as an n-gram model or a maximum entropy model (Jurafsky and Martin 2009; Hayes and Wilson 2008). However, beyond Gorman's analysis, which did not investigate the implications of different machine learning techniques on the issue at hand, none of the recent work has considered the nature of phonotactic knowledge and how modeling choices affect the conclusions which are derived from them.

When designing a model of phonotactic knowledge, multiple modeling decisions must be made, some of which are discussed in recent literature. First, does phonotactic knowledge apply to features, or does it apply only at the segmental level? It is quite surprising given the success of features in accounting for phonological patterns that recent work has shown through both machine learning and traditional computational models that training on segments provides better results and easier learning than training on features (Albright 2009; Mirea and Bicknell 2019).

Second, what is the appropriate representational unit, or window size, for onsets to be grouped in as models are trained? Per Gorman's 2013 model for monosyllabic nonce words,

a model of gross status treating each onset as a single unit suffices. [S T R] does not need to be recognized as a sequence of segments, but simply as a unit contained in a set of *attested onsets*.[3] However, machine learning models can easily accommodate the difference in window size as a model parameter, and this representation needs to be decided.[4]

Third, is phonotactic grammaticality a categorical or gradient measure? Both of these decisions will greatly impact model decisions and performance. Many available learning models can take any number of features as their inputs and could be adapted to a featural or segmental view, but the flexibility of some models for output as a binary, categorical classification, or gradient, is not as clear. Additionally, how well model output is analogous to a true probability is also murky (Hayes and Wilson 2008). I will discuss this issue with each model presented.

For my thesis, I do the following:

a. I implement four models:

   i. Gross phonotactic violation, following Gorman (2013);

   ii. A support vector machine, or SVM; a classical machine learning model frequently used for binary classification tasks.

   iii. A maximum entropy model as employed by Hayes and Wilson (2008) and referred to by phonotactic literature as a gradient model;

   iv. A recurrent neural network, or RNN; a deep learning model quite flexible to different configurations, and can be used to classify using a threshold or argmax over normalized outputs.

[3] Gorman (2013) acknowledges that this does not capture all phonotactic patterns, but operationalizing this model to include other aspects of phonotactics immediately becomes quite difficult and has not been explored.

[4] As an example, the onset[S T R] would be represented in the following ways for window size n: $n = 1$: ((S),(T),(R)), $n = 2$: ((ST), (TR)), and $n = 3$: ((STR)).

b. I compare each model's performance on a withheld test set to ensure that the model has made some generalizations about the data.

c. I then test each model's ability to categorize data from human judgment experiments collected by Scholes (1966), to see which approach is better to model human judgments.

d. I train well-performing models on datasets which have had the frequency of types equalized, and those which preserve the type frequency from the original data.

I train each model on a corpus of word onsets, controlling for three factors: First, previous literature has not been concerned with the appropriate context window size for phonotactic restrictions, so I train each model embedding the onsets by unigrams, bigrams, and trigrams. Second, I also train each model with both the full dataset, including all tokens taken from the dictionary, as well as the set of all onsets, removing any duplicates. This is to test if the models can learn which onsets are grammatical with only the set of attested and unattested onsets. Third, each model is trained only on segments, and not features. I do this following the work of Albright (2009) and Mirea and Bicknell (2019), whichs shows that phonotactic models trained on segments learn and perform better than models trained on features. I show that the maximum entropy model is able to correlate the most with human judgments and crucially that the RNN model as well as the maximum entropy model performance is not significantly affected when the phonotactic frequency differences are neutralized in the training data.

CHAPTER 2

RELATED WORKS

2.1 Origins of phonotactics

The study of phonotactics began with Halle (1962) and Chomsky and Halle (1965), where the aforementioned distinction of [blɪk] and [bnɪk] is introduced. This is a simple but powerful demonstration of phonotactic knowledge, which the authors define as restrictions on the underlying representation of words. These restrictions can be *segment structure constraints*, but also *sequence structure constraints* (Halle 1959). These are named *morpheme structure constraints* (MSCs) by the authors.

Segment structure constraints contain the constraints dictated by the available underlying segments in a given language. Though [ð, d] are both possible surface segments in Spanish, [ð] only appears as an allophone of /d/, thus a nonce word such as /ðano/ cannot appear as an underlying representation of a word in Spanish and violates the phonotactic system of the language. My thesis will not be concerned with this aspect of phonotactic knowledge, as it can be modeled quite simply in terms of set membership: Given a set of available segments S, for each x in sequence s, if $x \in S$, then no segment structure constraint is violated.

Sequence structure constraints are not so simply defined. Chomsky and Halle (1965) prefer a featural, rule-based constraint structure on underlying representations. An example of an English sequence structure constraint (from Gorman (2013)) is seen in (1).

(1) $\left[-\textsc{Cont}\right] \rightarrow \left[+\textsc{Liquid}\right] \; / \; \# \left[-\textsc{Cont}\right] \_\_\_\_\_$

This constraint prohibits a stop consonant after a word-initial stop, and prohibits the underlying */bnɪk/, but allows /blɪk/. A set of these MSCs then would represent language-specific phonotactic knowledge that is present in any given speaker.

If this knowledge is represented in such a way, it can be tested experimentally, which is the path Scholes (1966) took. Scholes (1966) used acceptability judgement tasks to probe these constraints and evaluate them. In his study (particularly, experiment 5 in his book), 33 seventh grade students were asked to judge whether each word in a list of nonce words was an acceptable English word or not. They were to give binary "yes" or "no" answers, and each word received the number of "yes" votes as its score. Examples of this data can be seen in Table 2.1. Nonce words are transcribed using the ARPAbet system[1], and with the corresponding IPA.

Word (ARPAbet)	word (IPA)	Number of "yes" responses
K R AH1 N	kɹʌn	33
F L ER1 K	flɚk	31
M R AH1 NG	mɹʌŋ	27
V R AH1 N	vɹʌn	19
N R AH1 N	nɹʌn	8
V P EY1 L	vpeɪl	0

Table 2.1: Examples of the nonce words used in Scholes' experiment, along with number of positive responses.

Scholes's data shows one problem of studying the categorical vs. gradient distinction. There is no way given this experiment to find gradience in the individual speaker judgments. Each speaker gave a categorical "yes" or "no" answer. Nevertheless, the proportion of "yes" to "no" answers can be split evenly in some cases, where the subjects in the study disagree with each other. The word V R AH1 N [vɹʌn], was only considered acceptable by roughly half of the subjects, for example.

The data from the Scholes (1966) experiments raises questions for a way to model phonotactics. The stimuli receive a range of responses that suggest even if a word is judged as unacceptable, some words might be *more* unacceptable than others. Incidentally, Chomsky and Halle (1968) acknowledge this; along with [blɪk] and [bnɪk] as examples to represent acceptable and unacceptable, they introduce [bznk], which they claim should be less accept-

[1]See Section 3.2 for more discussion on this system.

able than [bnɪk]. A theory that is simply a set of MSCs, designed to exclude */bnɪk/ but allow /blɪk/ does not express these levels of unacceptability.

These facts about the acceptability of various nonce words has led to the theory that the (phonotactic) grammaticality is gradient, and further evidence for this comes from experiments showing a spread of different levels of acceptability for nonce word rating tasks (Albright 2009; Hayes and Wilson 2008; Shademan 2006). These experiments find that participants always give gradient responses when given a scale as a method of response.

2.2 Nature of constraints and output

The central question raised by Gorman (2013) is the that of gradient vs. categorical knowledge. Under Gorman's view, three questions need to be addressed by any theory of gradient grammaticality phonotactics:

a. Do experiment participants have the ability to accurately perceive the grammaticality of a test item and map their acceptability judgment to that grammaticality?

b. Do intermediate acceptability judgments constitute evidence for gradient grammaticality?

c. Is the gradient theory compared to categorical alternatives in a way that clearly shows the advantages of the gradient theory?

Speakers have difficulty with perception of nonce words that have phonotactic violations (Dupoux et al. 2004; Kabak and Idsardi 2007), and sometimes repair them with illusory vowels dependent on native language phonology (Durvasula et al. 2018). If this is the case, speakers are giving an acceptability judgment on a perceived item that is different than the stimulus. A full theory of gradient phonotactics would necessitate accounting for phonotactic violation repairs systematically, so it is clear what the speaker is making their acceptability judgments based off of. Though Gorman (2013) points this out as a complication for gradient phonotactics, it should also be considered as a potential complication for a categorical theory

as well; a categorical judgment still relies on a perceived item that could exhibit differences from the intended stimulus.

Secondly, intermediate acceptability judgments do not constitute direct evidence for gradient grammaticality. Participants asked to make a judgment task on numbers presented to them with a scale of how "representative" those numbers were of "even" and "odd" use the intermediate measures even though there is a clear, categorical distinction between the two categories (Figure 2.1).

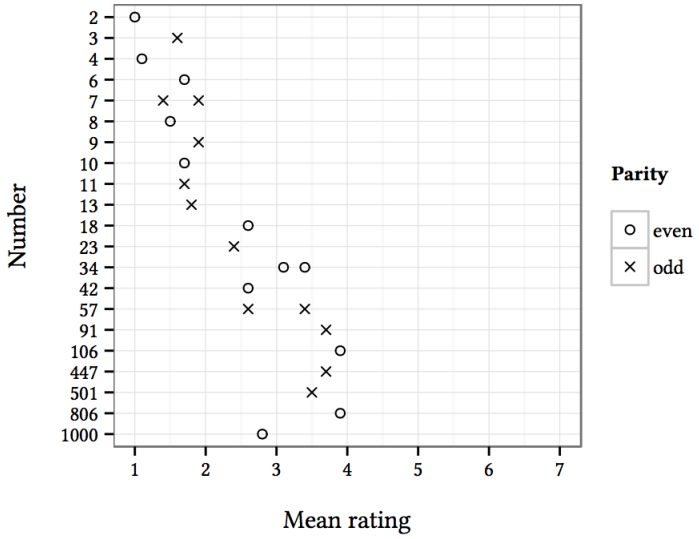

Figure 2.1: Mean ratings of subjects asked how representative the stimuli were of the categories "even" and "odd" respectively, where "1" indicates most representative (Gorman 2013; Armstrong, L. Gleitman, and H. Gleitman 1983).

Though there are potential explanations for why participants believe 23 is more odd than 57, Armstrong, L. Gleitman, and H. Gleitman (1983) point out that subjects' ability to compute with these numbers and explain the definition of odd and even numbers are the fundamental facts of the task, and a theory dedicated to explaining the gradient judgments will find it difficult to represent basic human knowledge about odd numbers. Gorman (2013) then suggests a parallel phonology example: orthodox beliefs about phonotactic acceptability dictate that acceptability is closely related to frequency; the more often a sequence appears

in a language, the more acceptable it is. For example, Pierrehumbert (1993) argues that there is a relationship between perceived well-formedness of a phoneme combination and its frequency in the language. However, if sequences [bl] and [kl] have different acceptability measures due to their frequency ([bl] appears roughly twice more than [kl]), how are they to be treated equally under independent phonological processes like syllabification, etc.?

Lastly, gradient models should be able to demonstrate an advantage over a baseline categorical model and address the challenges raised above. As stated, current methods assume the correlation of gradient acceptability and frequency of patterns in the lexicon is a causal relationship, but fail to test their models to see if learning is affected by removing frequency information from the model.

As mentioned above, two principled ways exist to represent phonological segments: as the discrete segments themselves or as vectors of phonological features. Hayes and Wilson (2008) use a featural system to train their model and do not compare model perforance with a segment-based model. This was addressed by the work of Albright (2009), and now Mirea and Bicknell (2019), who both show support for a segmental phonotactic system rather than a featural system. Because of this, I will choose to focus on the problems of gradience in the model structure and input, and train my own models using segmental representations.

2.3 Dealing with Gradient Acceptability: Recent Models

Now I will turn to a discussion of prior work conducted on gradient phonotactics models (Albright 2009; Bailey and Hahn 2001; Hayes and Wilson 2008; Mayer and Nelson 2019). Albright (2007) and Albright and Hayes (2003) conducted experiments where participants responded to nonce words on a 7-point Likert scale, with some variation of "completely impossible as an English word" on one end of the scale, and "would make a fine English word" on the other end. These studies highlight the existence of gradient acceptability: [bwɪk] is not as acceptable as [blɪk], but is more acceptable than [bnɪk]. Albright (2009) introduced a model of probabilistic phonotactics to account for these experimental results.

The model Albright (2009) used to account for the human acceptability judgments is an n-gram model, common in natural language processing for language modeling; it is a language model that predicts the most likely next segment given a preceding sequence of segments that can range in length (Jurafsky and Martin 2009). For example, a trigram model (*tri-* meaning the model predicts an upcoming segment based on the previous two segments) for the onset [spl], where # represents initial word boundary and N represents the syllable nucleus, would look like the following:

(2) $\hat{p}(spl) = p(s|\#\#) \cdot p(p|\#s) \cdot p(l|sp) \cdot (N|pl)$

The probability of the whole onset is modeled as the product of the sequential probabilities of each segment given the two preceding segments.

Another aspect that can be modeled as affecting the probability of a sequence is neighborhood density: what number of licit sequences exist that are one step away from the sequence in question, where a step can be either a single insertion, single deletion, or single substitution of a segment (Coltheart 1977)? Neighborhood density is a measure that is not directly phonotactic; but neighborhood density does have a high correlation with bigram probability (Bailey and Hahn 2001). Neighborhood density also has a clear and strong effect in longer nonce words with recognizable morphology, and even if these have strong violations of phonotactics (e.g. mrupation) they will receive English-like ratings (Hay, Pierrehumbert, and Beckman 2004).

Hayes and Wilson (2008) created a phonotactic learner based on *maximum entropy grammar* which they claim can correlate well with the gradience in human judgments (Goldwater and Johnson 2003; Jaynes 1983). They train their phonotactic learner on onsets, and use a *type* frequency structure in the training data. They claim to replicate gradience of human judgments by reporting correlation co-efficients with the results from Scholes (1966). As discussed, this makes an assumption of individual speaker judgment gradience based on the aggregate responses of the Scholes participants, when each participant gave a yes/no answer. Thus, it can be argued that Hayes and Wilson are really showing they can correlate

model predictions with the aggregate score of onsets in the pool of participants in Scholes' experiment. While I think this is an important point of discussion, I do not explore this in my thesis, assuming that the Scholes data is reasonably representative of individual speaker judgments.

Mayer and Nelson (2019) introduce a recurrent neural network language model (RNNLM) which performs better than the Hayes and Wilson phonotactic learner at correlating with judgment data. Their model is different from the RNN used in my work in that it is not trained and tested over onsets, but is rather trained on the whole words to learn probability distributions over the transitions between segments, and these probability distributions can be used to estimate the probability of a any string as a licit phonotactic sequence.

RNNLMs are more recently utilized models that have come out of the natural language processing literature in the last two decades due to the rise in increased computing power needed to train the models (Mikolov et al. 2010). Mayer and Nelson (2019) train RNNLMs on various corpora of several languages to see if these models can learn phonotactic phenomena in those languages. They test model performance in predicting vowel harmony rules in Finnish, Cochabamba Quechua laryngeal co-occurrence restrictions, and sonority sequencing rules in English. They find that the neural nets perform better than the maximum entropy model of Hayes and Wilson in correlating with human judgments. For English, Mayer and Nelson (2019) evaluate their model and the phonotactic learner on experimental results from Daland et al. (2011). This experiment was meant to probe judgments regarding the phonotactics of sonority sequencing, but was not compared to any other commonly used phonotactic judgments dataset.

2.4 Frequency structure in the data

A crucial part of probabilistic phonotactics is what frequency structure is important in learning. Models can be trained on either *type* or *token* frequency.[2] Hayes and Wilson (2008)

[2]The terms type and token need to be explained further in terms of onset frequencies. One can imagine that *token* frequency is the frequency of onsets in the corpus of speech or

argue that type frequency is what is more accurate for training data (as opposed to token frequency), and that is what they use as training for their phonotactic learner. As I have also done, the authors have removed what they term *exotic* onsets from the CMU dictionary and train the learner on a nonexotic corpus of onsets along with their type frequencies.

However, what if the frequency structure of types is not important at all? Again, many recent gradient models fail to adequately consider the assumptions made in abandoning a categorical model. I train the models I will be using on a corpus of onsets with type frequencies dictating their distribution in the training data, and then also train all of the models with the frequency of the onsets equalized across the training data. I do this to expand on the comments in Hayes and Wilson (2008) regarding the role of frequency structure in the training data to include approaches that still question the relationship between frequency structure and phonotactic acceptability.

I will refer to data with type frequency (the frequency that is present in the dictionary) as *type frequency data*, because there is a cline of onsets, from those that are marginally represented to those that are present in the thousands. Data where this frequency structure has been removed I will refer to as *equalized frequency data*. It is important to distinguish that this categorical vs. gradient distinction in judgments is not the same as the comparison being made in the frequency structure of the data. This frequency structure comparison can instead be explained as a deterministic learning method vs. a probabilistic learning method. If the frequency structure is integral to proper acquisition of phonotactics, this should be reflected in the ability of the model to predict human judgment.

Though the distinction between judgments and learning should be made, phonotactic knowledge built on the probability distribution of possible onsets suggests that the knowl-

text, and this would lead to an high frequency for ð, for example, since this is a common onset in highly occuring function words (e.g. the, that, these, those, this). Of course, this onset actually has very low occurences in the lexicon at large, and a count of occurences in the CMU dictionary would yield a count of *types* and not *tokens*, since what is being counted is the number of words in the lexicon that exhibit the onset and not the number of words in some naturalistic corpus.

edge must be gradient in nature, whereas phonotactic knowledge built on a set of attested onsets suggests that the knowledge is categorical in nature. This is what links the frequency structure of the data with the nature of phonotactic judgments.

CHAPTER 3

METHODS

3.1 Phonotactic models for this study

For this study I will compare my own implementations of machine learning models and compare them to other computational models in the literature. They will be fitted with word onset data from the Carnegie Mellon University (CMU) Pronouncing Dictionary (Weide 1998). I use this data because the stimuli are designed to test the phonotactic acceptability of the nonce word onset only, outfitted with a set of rhymes that have only simplex codas and are all attested. Other available data (Albright and Hayes 2003; Albright 2009) commonly used for evaluations (Gorman 2013; Mirea and Bicknell 2019) do not follow this with their stimuli, using complex codas and much more variability in their experiments. Due to this, these experiments are not suitable for evaluation of models trained only on onsets, because participant responses are affected by the presence or lack of phonotactic violations in the rhyme.

As is standard practice in machine learning, models will be evaluated based on a withheld test set of data removed from the dataset before training, and then used to predict the data of Scholes (1966) This practice has been absent from much of the phonotactics literature, with models trained and tested on the same data, or tuned to the test set of data in some way (Hayes and Wilson 2008; Goldwater and Johnson 2003). The reason this should be avoided is that it does not allow for sufficient generalization from the training. The goal is to produce a model that can adequately account for unseen data, but if the model is evaluated on the data it learned from, it will produce artificially accurate results.

3.2 Data Preparation

The CMU Pronouncing Dictionary is an open-source pronunciation dictionary of North American English. It contains roughly 134,000 words and their pronunciations. Pronunciations are transcribed using the Advanced Research Projects Agency phonetic transcription codes, commonly known as the ARPAbet system. The CMU dictionary uses 39 ARPAbet phonemes as well as primary and secondary stress markers to transcribe entries.

For data preprocessing, the dictionary entries and their pronunciations were imported to Python (Python Software Foundation n.d.), using the Pandas library DataFrame object (McKinney 2010). The 160 unique onsets are then isolated for analysis, and *padded* to the length of the longest onset. This padding is used because the models require inputs of a fixed length, and the standard natural language processing procedure is to add some null character to pad any piece of data shorter than the longest item needed. In the case of onsets, the longest onsets in English have a length of three, so those can be represented as [S P L], whereas a simplex onset is represented with two added null characters as in [# # B].

Some of the onsets in the CMU dictionary occur only once or a few times. I examined these onsets and, based on my judgment as a native speaker, decided that they are not attested in my own lexicon, and thus they are not representative of the phonotactic knowledge I want to model, and should be left out for analysis. I removed any unique onset which occurs less than 35 times in the dictionary. To decide this cutoff point, I examined the onsets manually to see if the low-frequency onsets were indeed unacceptable for my judgment, and found that removing onsets with a frequency of less than 35 removed all the onsets I found unacceptable. Examples of such onsets are [# Z B] and [# HH M]. Hayes and Wilson (2008) employ this same method, whereas Gorman (2013) opts to reduce the onsets represented in the dictionary by eliminating each word that has a frequency less than one per one million words in the SubtlexUS corpus, a dictionary containing word frequencies in American English (New and Pallier 2009).

Negative data is generated by permuting all possible consonant combinations of lengths 1

Onset	Frequency	Onset	Frequency	Onset	Frequency
no onset	20,285	k	13,042	s	12,571
b	9,777	m	9,547	p	7,866
d	7,779	r	7,483	h	6,734
f	5,633	l	5,511	g	4,986
t	4,941	w	3,864	n	3,429
ʃ	2,496	v	2,445	st	2,362
dʒ	2,182	pr	1,796	br	1,607
kr	1,514	j	1,383	gr	1,285
tʃ	1,278	tr	1,197	kl	999
sk	984	z	959	sp	915
fr	914	bl	705	θ	651
fl	651	pl	528	dr	483
kw	471	str	460	sl	408
gl	393	sw	384	hw	367
sm	249	sn	244	kj	191
ʃr	164	hj	163	skr	153
bj	149	mj	142	tw	123
spr	121	θr	114	fj	107
ʃw	101	ʒ	99	gw	86
skw	85	pj	84	ʃl	81
ð	72	ʃm	66	ʃn	57
dw	44	spl	40		

Table 3.1: Onsets used for training

to 3, and subtracting from these the set of unique onsets found in the CMU dictionary. This resulted in a set of 12,500 negative onsets. Each positive example is left at its frequency in the CMU dictionary, totalling 128,000 tokens of positive onsets. In order to generate the equalized data, the positive data is reduced to a set of all onsets represented in Table 3.1, and the set is multiplied to reflect the size of the data with frequency information.

For example, consider a made up language where there are 100 unique words and 5 unique onsets = sn, spl, r, m, j, and this data is used to train a model that requires of at least 100 data points. The onsets could be structured in the training data proportionally to their appearance in the lexicon of this made up language, or they could be equally represented at a number large enough to train the model, as in Table 3.2.

Onset	Gradient data count	Equalized data count
sn	40	20
spl	25	20
r	15	20
m	12	20
j	8	20
Total count	100	100

Table 3.2: Toy example of different frequency structures in training data.

3.3 SVM model

SVMs are supervised learning models and are powerful, easy-to-use, discriminative binary classifiers. These facts make them well-suited for baseline classification and for categorical classification tasks. Standard SVM models do not provide a probability as an output, because they only categorize the data as being on one side of a dividing hyperplane, and distance from the hyperplane cannot correlate with probability of class membership.

The SVM model takes each data point as an n-dimensional vector, and seeks to find the hyperplane in $n-1$ space such that the distance between the hyperplane and the nearest data points, referred to as the support vectors, of both classes is maximized. An example of such a hyperplane is provided in Figure 3.1, where two classes are separated by a hyperplane (which in 2 dimensions is just a line). Supporting packages in the Scikit-Learn library in Python (Pedregosa et al. 2011) allow for the onsets from the CMU dictionary to be vectorized and fed into the SVM. Each vector has length n where n is the number of segments in the data, and the vector for a given onset has values for 1 for each position correlated with the segments in that onset.

How is this hyperplane determined from the training data? If the training data is a set of pairs x, y such that $x^{(i)}$ is a vector for the i^{th} data point in the training data and $y^{(i)}$ is the value for that data point ($y \in 1, -1$) where we can use 1 to represent attested data and -1 for the generated unnattested data. With this form, the classifier looks like the following:[1]

[1]This is in a sense more deterministic than logistic regression or neural nets, as will be

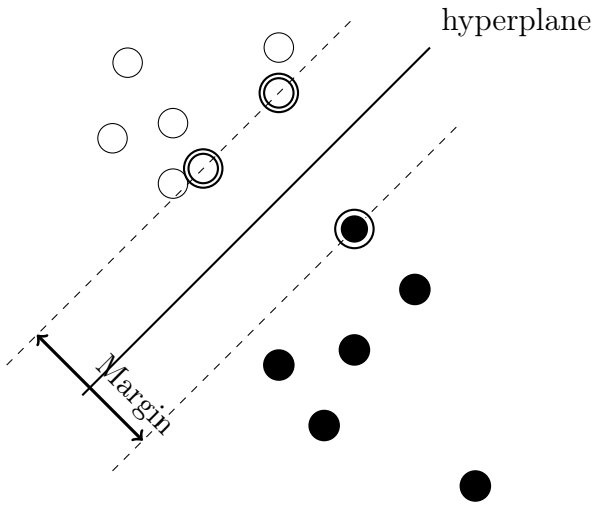

Figure 3.1: Illustration of an SVM in 2 dimensions

(3) $h_{w,b}(x) = g(w^T x + b)$

where

$g(z) = 1$ if $z \geq 0$, $g(z) = 0$ otherwise

In (3), the parameters (w, b) represent some hyperplane. And given the training pair $(x^{(i)}, y^{(i)})$, the *functional margin* of (w, b) is as in (4).

(4) $\hat{\gamma}^{(i)} = y^{(i)}(w^T x + b)$

This functional margin value represents the distance between the hyperplane and the data point and should be large to reflect a confident prediction far away from the separation line. So in order to attain a large functional margin for $(x^{(i)}, y^{(i)})$, if $y^{(i)}$ is negative, $(w^T x + b)$ should be a large negative number, and if $y^{(i)}$ is positive, $y^{(i)}$ should be a large positive number. The functional margin value should always be positive (if it is negative, it is on the wrong side of the hyperplane) and this is reflected in that if $h_{w,b}(x^{(i)}) = y^{(i)}$, then $y^{(i)}(w^T x + b) > 0$.

discussed in the next section. This is because the SVM classifier directly predicts the class of the input, whereas in logistic regression, there is an intermediate step of estimating the probability of class membership before classification.

Though this is for only one data point, this process can be expanded to an entire set of data where the functional margin with respect to the set is the smallest of the functional margins for the individual data points. These data points are the support vectors.

(5) Given training set S:

$S = (x^{(i)}, y^{(i)}), i = 1, ..., m$

then

$\hat{\gamma} = \min_{i=1,...,m} \hat{\gamma}^{(i)}$

Lastly, the margin needs to be maximized. A number of hyperplanes can be drawn that fail to maximize the margin between the hyperplane and the support vectors. However, the function (4) can be made arbitrarily large because the classifier (3) cares only about the sign of $(w^T x + b)$, not the magnitude.[2].

In order to do this, the functional margin can be divided by the euclidean norm of the vector w, which normalizes the margin and assures the margin is not maximized artificially through the classifier parameters. Putting this all together[3], the maximization function is as in (6):

(6) $\max_{\hat{\gamma},w,b} \frac{\hat{\gamma}}{\|w\|}$

where

$y^{(i)}(w^T x + b) \geq \hat{\gamma}, i = 1, ..., m$

For my model, I am using the `sklearn.svm.SVC` class, with the `kernel` parameter set to `"linear"`. I ran the SVM three times, testing the best unit to vectorize the data over. In order to train the SVM, the data must be embedded numerically, and this can be done by counting unique unigrams (the segments themselves), unique bigrams, (pairs of segments), or trigrams (triads of segments). and unigrams perform best by a slight amount.

[2]For example, $g(w^T x + b) = g(2w^T x + 2b)$

[3]This overview of SVM classifiers is greatly simplified and is only meant to provide an intuition of how the functions are conceptualized. In reality, the function in (6) is non-convex and quite difficult to solve and more steps are necessary to implement this classifier from scratch.

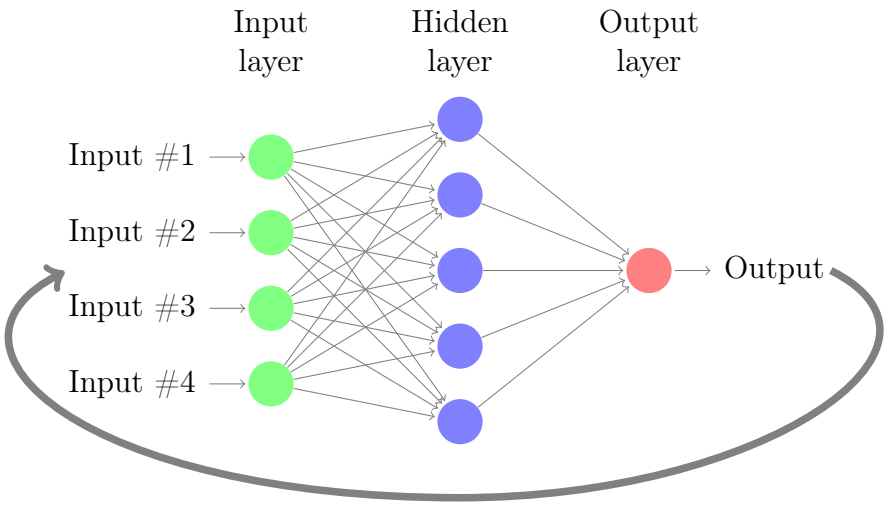

Figure 3.2: Sketch of the input nodes, hidden layer, and output of a Recurrent Neural Network.

3.4 RNN model

The probabilistic model I am using is a Recurrent Neural Network (RNN), which is a simple neural net well-suited to sequential data (Elman 1990)[4]. The RNN is a network of nodes and edges, with layers of nodes, and edges between each layer. Each node and each edge can have a corresponding number associated with it, which is added or multiplied to the input and sent to the next layer as its input. The input layer of size n can take an n-dimensional vector, traverse through the layers to the output layer, where the output layer is fed back in to the input of the next iteration. This allows for each segment in the onset to be embedded as a number and fed into the network one at a time. Each iteration, the network makes a preliminary guess, and this guess is fed into the network again while it simultaneously analyzes the next segment. Thus the final output of the network takes into account the sequential information of the onset and does not treat it as a bag of segments. A simplified visualization of the network architecture is available in Figure 3.2.

I am using an input layer n where n is the number of unique values in the data. For unigrams, this is the number of consonants, for bigrams, the number of unique bigrams, etc.

[4]Model implementations for the SVM and RNN are documented at https://osf.io/f76zn/ (Sarver 2020).

I am using a hidden layer size of 128. When the network is being trained, a loss function calculates how far off the model's guess is after each iteration, and an optimization function uses that information to traverse backwards through the network and update all the weights and biases accordingly. I am using the Cross-Entropy Loss function and Stochastic Gradient Descent for optimization. Training is done over 50,000 iterations, where for each iteration, a random training pair is selected from the training data and fitted to the model.

The model is implemented in the `PyTorch` framework (Paszke et al. 2017). The model has the same accuracy level regardless of whether the data is embedded in unigrams, bigrams, or trigrams, so the RNN model is easily able to classify these onsets regardless of the embedding strategy. Results discussed in my thesis are from the model iteration that uses the bigram embeddings. This model runs the fastest due to the least amount of parameters and relatively small input layer.

The output layer consists of two nodes, one representing a valid segment and one representing an invalid segment. The output values are plugged into a sigmoid function (seen in (7)) which places the values on a scale between 0 and 1. This function is displayed visually in Figure 3.3. An extremely high value will be placed at 1 on the scale and represents that the model has 100% confidence in that output, an input value of 0 is placed at 0.5 and represents 50% confidence, an extremely low value represents 0% confidence. The node with the higher value is selected as the model's predicted onset.

$$(7) \quad \sigma(x) = \frac{1}{1+e^{-x}}$$

The neural net model has structural differences to the SVM that require extra care in determining how the data is fed to the model. While the SVM can receive the entire onset as an n-dimensional vector, the recurrent neural net model is sequential; each segment of the onset is fed into the model sequentially. This can of course also vary by window size: for the onset [str], this sequence can occur over single segments, segment pairs, or segment triplets. This gives the RNN more power to represent the dependencies between segments and the effects they have on acceptability.

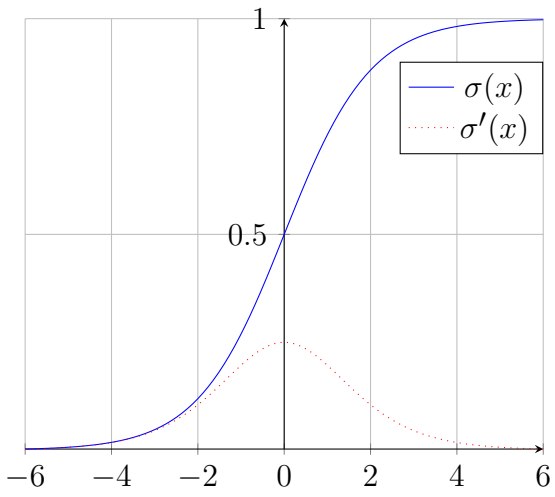

Figure 3.3: Graph representation of the sigmoid function, with x = input and y = output.

3.5 Maximum Entropy grammar

A maximum entropy model, notably used by Hayes and Wilson (2008) to build a phonotactic grammar, is also suited to providing a probabilistic output. This model has also been used in phonology as a method of learning OT constraints (Goldwater and Johnson 2003). A maximum entropy model can express the probability of member x to the set of possible forms Ω. Given a set of observed data, the learning algorithm generates a model of constraints and weights each constraint such that the probability of observed data is *maximized*.

The resulting model will look like this toy example provided by Hayes and Wilson (2008) in Table 3.3. Assume that the grammar has two constraints, *#V and *C#, that have the weights 3 and 2 respectively. This grammar would assign different maxent values to the lexical items CV, CVC, and V. CV does not trigger any violation, so its rating is highest, followed by CVC, and C.

x	*#V ($w = 3$)	*C# ($w = 2$)	Score ($h(x)$)	Maxent value ($P^*(x)$)
CV	$3 \cdot 0$	$2 \cdot 0$	$(3 \cdot 0) + (2 \cdot 0) = 0$	$\exp(-0) = 1$
CVC	$3 \cdot 0$	$2 \cdot 1$	$(3 \cdot 0) + (2 \cdot 1) = 2$	$\exp(-2) \cong 0.14$
V	$3 \cdot 1$	$2 \cdot 0$	$(3 \cdot 1) + (2 \cdot 0) = 3$	$\exp(-3) \cong 0.05$

Table 3.3: Maxent Grammar (note: '·' represents multiplication here)

Assume briefly that the constraints and weights are optimized for the observed data. The model sums the product of constraint violations and weights to achieve a score, which is expressed as in (8):

$$(8) \quad h(x) = \sum_{i=1}^{N} w_i C_i(x)$$

Here, w_i represents the weight of the ith constraint, C_i represents the number of violations of that constraint, and $\sum_{i=1}^{N}$ represents the sum *maxent value*, and is calculated with (9). This is not a probability, but rather demonstrates the relative probability of the input.

$$(9) \quad \boldsymbol{P}^*(x) = e^{-h(x)}$$

And probability is calculated with the maxent value in (10):

$$(10) \quad P(x) = \boldsymbol{P}^*(x)/Z$$

where

$$Z = \sum_{y \in \Omega} \boldsymbol{P}^*(y)$$

The output used in phonotactic learning is not the probability of the input itself, which due to the large number of possible forms contained in Ω is impractical to report. Rather, the maxent value meant to show the relative probability between the forms, is given.

How are the constraints and weights for the model determined? The model name refers to its function of maximizing the *entropy*, a measure of randomness in the system[5], which S. Della Pietra, V. Della Pietra, and Lafferty (1997) show is equivalent to maximizing the probability (see (11)) of the observed forms.

$$(11) \quad P(D) = \prod_{x \in D} P(x)$$

where

P = set of observed data

This probability is maximized by an iterative search algorithm similar to the stochastic gradient descent described in the discussion of neural nets. All constraint weights N and

[5] $-\sum_{x \in \Omega} P(x) log(P(x))$, Cover and Thomas (1991)

total probability create a surface in $(N+1)-$dimensional space, and though the surface is never calculated as a whole, at each stage the local gradient is determined and the search iterates upwards (in the direction of higher total probability of observed forms) until a maximum is reached. Unlike neural nets, this surface is always convex, without only one global maximum for the search to find (S. Della Pietra, V. Della Pietra, and Lafferty 1997). The specific algorithm used by Hayes and Wilson (2008) in their phonotactic learner is the conjugate gradient method (Vetterling et al. 1992).

Hayes and Wilson (2008) set up their learner to maximize $log(P(D))$ for mathematical convenience, since the log function is monotonic and adjusting the weights to maximize $log(P(D))$ will necessarily maximize $P(D)$. The partial derivative of each weight $\frac{\partial}{\partial w_i} log(P(D))$ expresses the rate $log(P(D))$ will change in relation to that weight w_i, and the gradient is a vector of these partial derivatives. According to S. Della Pietra, V. Della Pietra, and Lafferty (1997), $\frac{\partial}{\partial w_i} log(P(D))$ is additionally interpretable as the difference between observed violations of constraint C_i and expected violations of the constraint, formally $O[C_i] - E[C_i]$.

Calculating $E[C_i]$ necessitates a limit on the length of forms in Ω (all possible forms for our model; in my case, all logically possible onsets), otherwise the set is infinite. In accordance with other models used in this thesis, I limit all forms in Ω to a length of three segments or less. Now that Ω is a finite set. $E[C_i]$ is expressed in (12):

(12) $E[C_i] = \sum_{x \in \Omega} P(x) C_i(x)$

where

$P(x) =$ probability of x

$C_i(x) =$ number of C_i violations by x

Only one more piece is needed before presenting the full learning algorithm, which is a measure of *accuracy* for the constraints. This accuracy measure calculates how the ratio of observed constraint violations ($O(C_i)$) with expected constraint violations ($E(C_i)$), or O/E.

Hayes and Wilson (2008) also implement a statistical upper confidence limit on O/E to reflect a difference in accuracy between an O/E that equals 0/10 and one that equals 0/1000. Effectively, this means that instead of the accuracies being both 0 and zero, 0/10 has 0.22 accuracy score and 0/1000 has 0.002. This is because if there are only 10 logically possible violations, a low number of observed violations does not imply as strong of a constraint as if there were 1000 logically possible violations.

With these pieces, the learning algorithm (Hayes and Wilson 2008) is constructed as follows:

(13) *Phonotactic Learning Algorithm Input*

A set Σ of segments classified by a set $\mathscr{F}$ of features, a set $\mathscr{D}$ of surface forms drawn from $\Sigma*$, an ascending set $\mathscr{A}$ of accuracy levels, and a maximum constraint size $\mathscr{N}$

Algorithm 1 Phonotactic Learning Algorithm

```
1: procedure PhonotacticLearner(𝒜, 𝒟, ℱ, 𝒩, Σ)
2:     Initialize empty grammar 𝒢
3:     for each accuracy level a in 𝒜 do
4:         while Exists constraint with accuracy < a do      ▷ Constraints by 𝒟, ℱ, 𝒩, Σ
5:             select the most general constraint and add it to 𝒢
6:             train the weights of the constraints in 𝒢                ▷ Gradient ascent
7:         end while
8:     end for
9:     return 𝒢                                              ▷ In the form of Table 3.3
10: end procedure
```

CHAPTER 4

RESULTS

Two main metrics are used to report the results of the models. The RNN and SVM are both classifiers, but with slight distinctions: the SVM output assigns a direct binary label predicting that the input is attested or unattested to its input, whereas the RNN assigns a score between 0 and 1, then assigns a binary label based on whether that score is above or below the threshold value of 0.5. For these classifiers, one metric is the *accuracy* of classification. Accuracy is the ratio of onsets whose classification correctly matches whether that onset is truly attested or not (gross status).

Accuracy is collapsed across classes, so for unbalanced data sets, if one class has much more data than the other and that class is predicted correctly, the accuracy can be very high even if the smaller class is poorly predicted. For this reason, I will break down the results into confusion matrices which display the results by class and will be explained further.

Ultimately however, the human acceptability judgments are not predictable by classification since they are gradient. To compare model output to these judgments, I will use Pearson's r correlation coefficient, which is a measure of linear correlation between the model output and the judgment data. A correlation of 1 means that the model perfectly predicts each score (in this case plotting the results would result in all the points falling along a straight line). A correlation value of 0 means that there is no relationship between the model predictions and the human judgment data.

4.1 Gross Phonotactic Violation

The simplest way to account for phonotactic acceptability is that onsets that are attested in the lexicon are judged highly and those that are unattested receive a lower acceptability judgment. With respect to the modeling of onsets and Scholes (1966) data for this study, *attested* is defined as present in the set of onsets described in (3.1). The distribution of

ratings for attested and unattested onsets is shown in Figure 4.1. Both the attested and unattested classes show a concentrated distribution around the high and low Scholes ratings, respectively, with thin tails representing outliers. Onsets that were rated as acceptable by a large number of Scholes participants that were unattested were *mr* and *ʃl*, and onsets that were rated as acceptable by a low number of partipants that were attested were *ʃm* and *sf*. The correlation value (Pearson's r) for gross status and the Scholes data is 0.803.

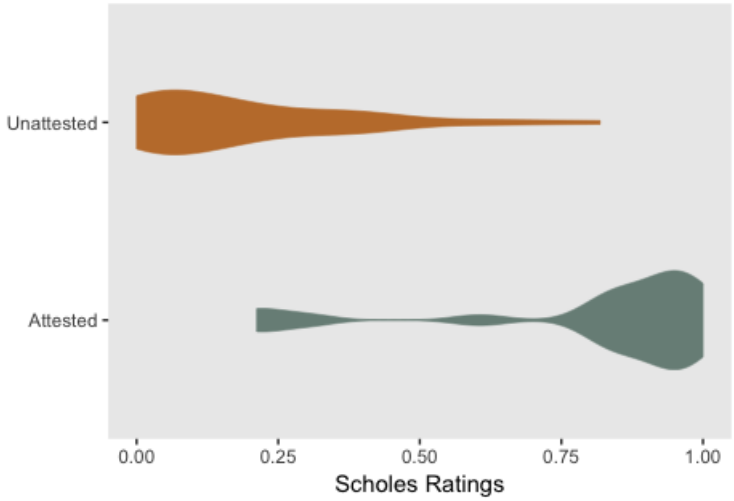

Figure 4.1: Distribution of normalized ratings in the Scholes experiment for attested and unattested onsets.

4.2 SVM results

Each model performs above 90% accuracy regardless of how the data is embedded, but a window size of one (e.g. an onset *spl* is represented as `((S), (P), (L))`) does lead to the best performance by a slight amount. See Figure 4.2 for comparisons.

Though the SVM does present a good model for separating categorical data, it is not well-suited to this task for a few reasons. Consider figures 4.3 and 4.4. They show a matrix of probabilities where the upper left quadrant is the probability that the model is given a negative onset and it guesses correctly, and the lower right quadrant is the probability that the model is given a positive onset and it guesses correctly. The lower left quadrant shows the probability of false negatives, where a positive data point is falsely classified as negative

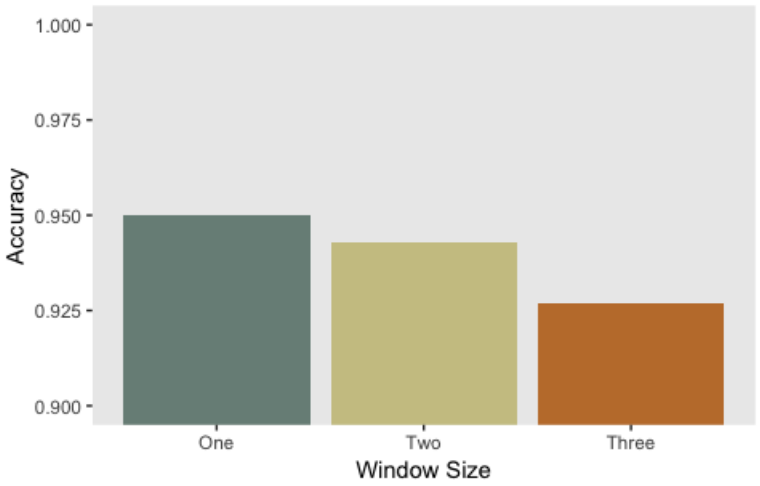

Figure 4.2: The accuracy of the SVM model's classifications of the test data based on the type of embedding.

by the model; and the upper right quadrant shows the probability of false positives, where a negative data point is falsely classified as positive by the model.

The imbalance of negative and positive examples in the training data skew the model towards modeling of the negative examples. Looking at high rate of false positives, the model significantly under-performs in predicting positive examples. The fact that the training data is skewed towards the negative data likely plays a role in this. The high rate of false positives also holds in the predictions for the Scholes data when evaluated against the gross status of the stimuli.

When correlated with the ratings of the participants, the Pearson's r is 0.328. Though the SVM does not get a good result when compared against acceptability data, it is important to note that this does not discount a categorical model, given the performance of the gross phonotactic violation model ($r = 0.803$). Perhaps this rather has to do with the unbalanced data and the nature of SVM training, which will be discussed.

4.3 RNN results

The RNN performs quite well on the classification task, with accuracy above 92% for all iterations. The RNN outperforms the SVM and is also more robust to different ngram

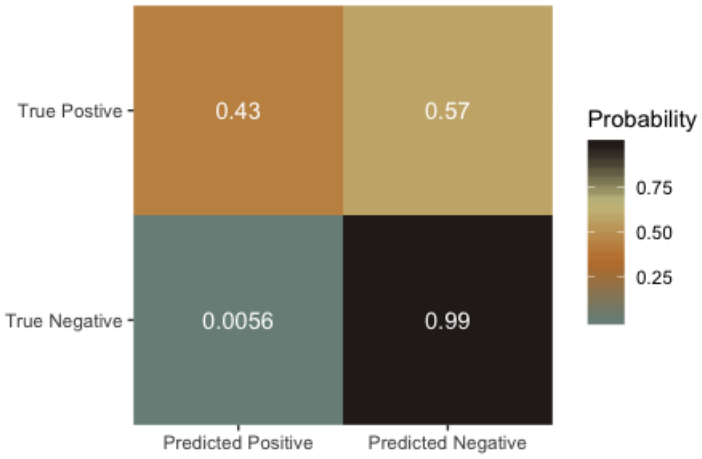

Figure 4.3: SVM results on withheld test set

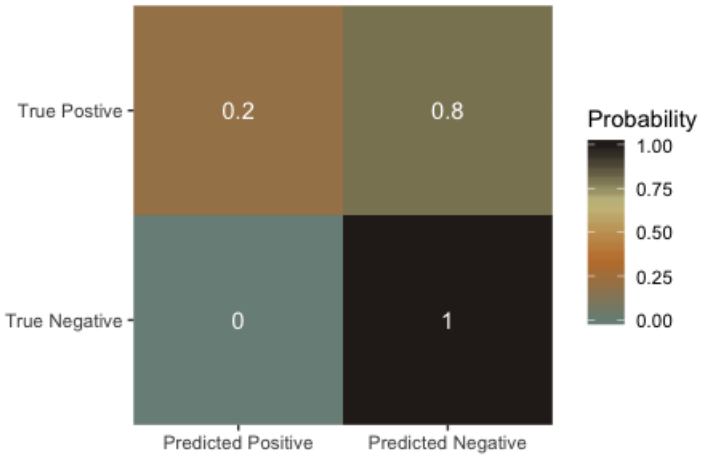

Figure 4.4: SVM predictions for Scholes data, with ground truth set to gross status

embeddings, but the differences are minimal. The RNN learns very quickly, within the first 10,000 iterations. In Figure 4.5, the loss value is plotted over the iterations of the model. For each iteration, a loss value is calculated which measures the distance of the model confidence in the output from the desired output. The higher the value, the farther away the model is from the desired predictions.

Figure 4.6 shows a matrix of probabilities like discussed above, where the upper left quadrant is the probability of true positives, and the lower right quadrant is the probability

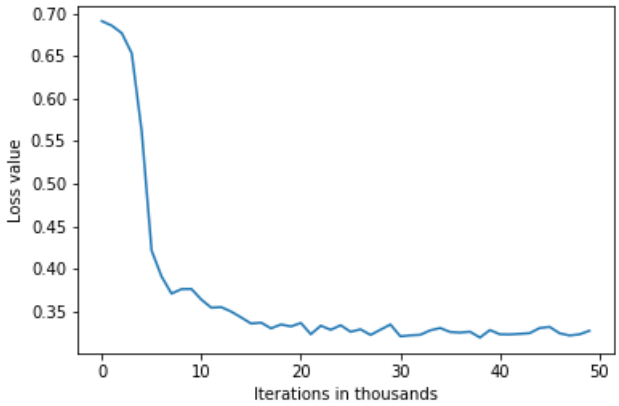

Figure 4.5: Loss values during RNN training, showing how wrong the model is for each iteration.

of true negatives.

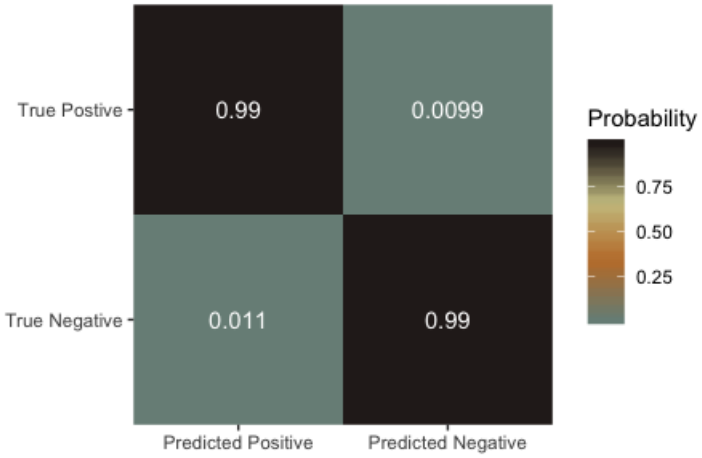

Figure 4.6: Confusion matrix showing the model's accuracy for guessing each class, and its error.

When the Scholes Experiment 5 data is compared against the model predictions, the neural net can categorize it with 81.4% accuracy (48 out of the 59 tokens). The RNN's confidence that an onset is grammatical is part of its output, and can be compared to the percentage of subjects that judge an onset as grammatical. This comparison is shown in Figure 4.7.[1]

[1]Note that some data points are overlapping in the figure.

The figure shows the onsets plotted based on the model's % confidence that the onset is grammatical (coded as having a value of 1), or ungrammatical (coded as having a value of 0), and the corresponding normalized judgments from the Scholes (1966) experiment. If the normalized judgment is 1, that means all 33 participants selected "yes" when asked if an onset was grammatical, and if the normalized judgment is 0, none of the 33 participants selected "yes." The onsets are also color-coded for whether they are actually attested onsets in English or not. The plot shows where the model is misclassifying onsets, and also shows that the model output is strictly falling only on the ends of the scale.

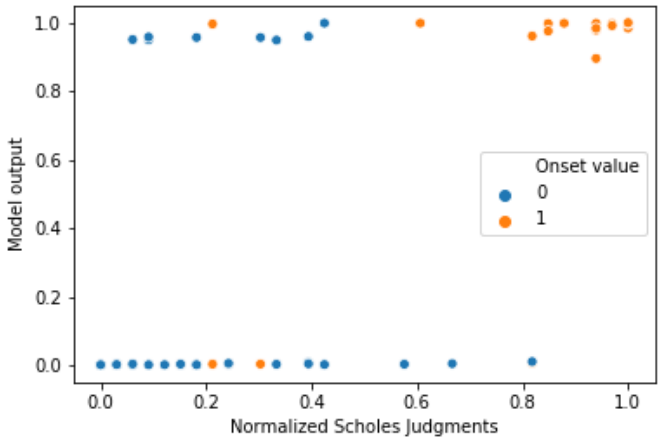

Figure 4.7: Scatter plot showing the model's % confidence that an onset is grammatical (value=1), compared to the percentage of "yes" responses in the Scholes experiment.

4.4 Changing frequency structures in the data for RNN training

When the RNN is trained on both type frequency and equalized frequency datasets, the accuracy remained similar in both cases.[2] For both withheld test sets, the precision is higher than the recall. Both models perform better at identifying positive cases than negative cases on the test data. Ultimately, the model trained on the equalized frequency dataset lags in

[2]Recall that the training data can represent the onsets using the frequencies with which they appeared in the CMU dictionary (gradient data), or represent each onset an equal amount of times (equalized data).

accuracy at 91%, while the type frequency model accurately classifies 94%. However, for the Scholes results, the models differ more. The model with gradient training has a higher precision than recall, but the model with equalized training has a perfect recall and lower precision, which leads to a higher overall accuracy in predicting the Scholes results. In terms of correlation, on the other hand, the correlation value comparing model predictions to the participant responses is significantly higher for the type frequency model. Ultimately, this is the more important measure as it reflects the model prediction of human responses.

Confusion Matrix	
0.99	0.01
0.11	0.89
Total Accuracy	
94%	

Table 4.1: Type frequency model results on withheld test set

Confusion Matrix	
0.98	0.02
0.16	0.84
Total Accuracy	
91%	

Table 4.2: Equalized Frequency Model results on withheld test set

Confusion Matrix	
0.83	0.17
0.28	0.72
Total Accuracy	
78%	
Pearson's r	
0.635	

Table 4.3: Type frequency model results predicting Scholes data

Confusion Matrix	
0.63	0.37
0.00	1.00
Total Accuracy	
82%	
Pearson's r	
0.458	

Table 4.4: Equalized frequency model results predicting Scholes data

4.5 Changing frequency structures in the data for MaxEnt training

As discussed before, the RNN's output exhibits little gradience. The model depends on thousands of iterations of training that reward confident predictions. It is worth comparing to

the maximum entropy phonotactic learner model (Hayes and Wilson 2008) which can provide a more gradient output. It is important to note that Hayes and Wilson use the following method to make their model output proportional to the normalized Scholes judgments: first, computing a *maxent value* from the model's score of a test item, as in (14), and then incorporating a free parameter T which is tuned to the test data to maximize the correlation values (15).

(14) $\boldsymbol{P}^*(x) = e^{-x}$

(15) $\text{predicted-rating}(x) = \boldsymbol{P}^*(x)^{1/T}$

The tuning parameter is a value that can be freely changed to morph the data into values with the highest correlation to the evaluation data. This is because the parameter can warp and expand the existing differences between the data to a larger scale or desired shape. With this parameter, the model can achieve a high correlation regardless of the structure of the training data, because it can warp the data so that judgments are more evenly spread through the intermediate range (See Figs. 4.8, 4.9, 4.10, and 4.11). When it is removed, the correlation falls slightly but both models are still relatively well-performing.

However, it is important to note that using such a parameter goes against standard machine learning practices. Because this tuning parameter is optimized to whatever data is being used to evaluate the model, it will not generalize well to predicting new data. If the goal is defining a general model of phonotactics that can correlate well to new human judgments, the model without the tuning parameter must be used. In Table 4.5, the correlations between the Maxent model results and the Scholes data show how the correlation drops when the tuning parameter is removed, and how the model predictions are extremely similar regardless of the training data used.

	Equalized frequency data	Type frequency data
Score with tuning	0.861 (T=10.05)	**0.880** (T=4.90)
Score without tuning	0.761	**0.769**

Table 4.5: Correlations of the Phonotactic Learner scores with the Scholes data

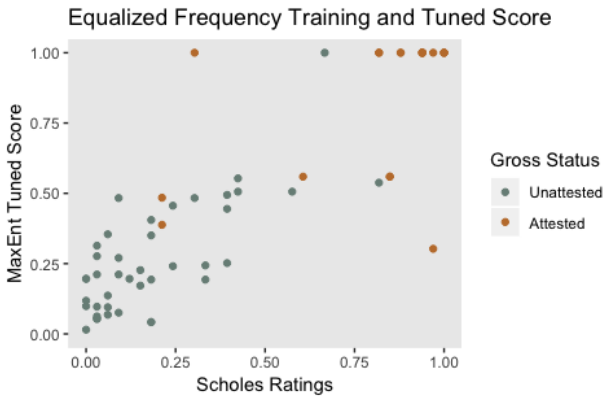

Figure 4.8: Equalized Frequency, Tuned

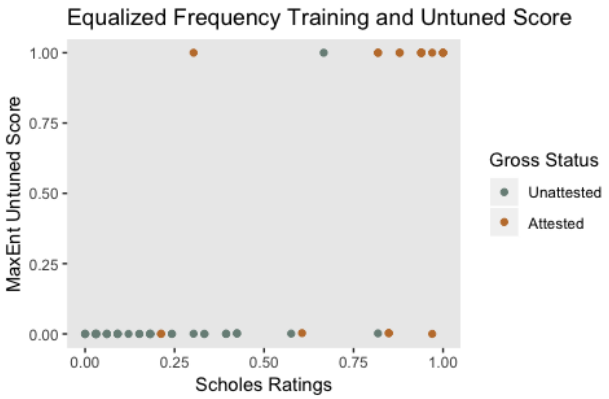

Figure 4.9: Equalized Frequency, Untuned

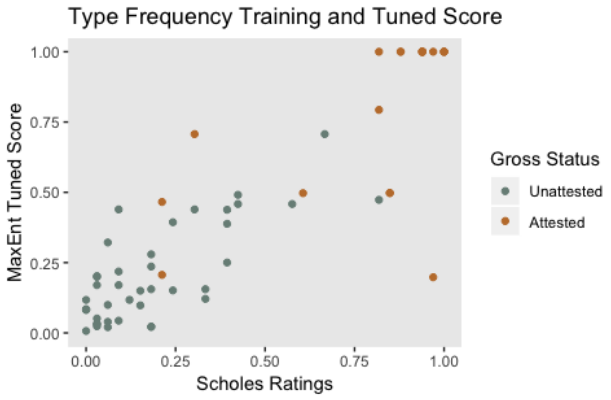

Figure 4.10: Type Frequency, Tuned

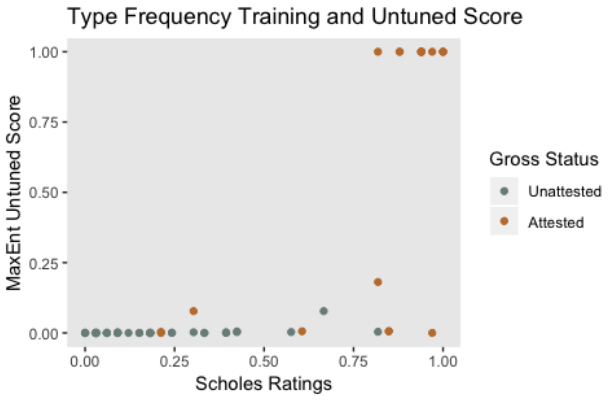

Figure 4.11: Type Frequency, Untuned

CHAPTER 5

DISCUSSION

5.1 Model Comparisons

Both the existing model of Hayes and Wilson (2008) and the baseline case of gross phonotactic violation perform better at correlating with the experimental data of Scholes (1966) than either the SVM or the RNN. However, it is insightful to note that with the removal of the tuning parameter, the plotted predictions of both the phonotactic learner and the RNN are strikingly similar. (See figures 4.9, 4.11, and 4.7). Specifically, these plots show a distribution of predicted data that falls in two groups at the top or the bottom of the plot, without any intermediate data in between. Though in theory the models can output any value between 0 and 1, the model ouput is strikingly binary. In the RNN's case, the model is trained to make binary judgments, and penalized for intermediate judgments. However for the phonotactic learner, there is no built-in operation that trains the model in this direction.

For the models presented in this paper, none exceed the correlation value achieved by correlating gross phonotactic violation with the Scholes data without tuning. This is an important comparison to make; though certain amounts of manipulation might produce a gradient model that also achieves a high correlation value, to what degree that model is evidence for underlying gradient grammaticality depends on the success of alternative explanations. If the gross status of phonotactic sequences can explain judgment data with a similar level of success as a proposed gradient model, no one model can be claimed as evidence for the nature of the human behavior that the models are explaining. All that has been shown is that models with certain assumptions about gradience can also explain the data in some way. These models are still far away from providing evidence for the nature of phonotactic grammar.

It is also important to consider that for evaluating these models, the data from Scholes

(1966) is not an extensive test set; it simply represents 60 onsets where participants could only choose a yes/no answer. A future step is to continue to evaluate these models against human judgments in different experimental settings to assure that these models are truly correlating with human judgments.

Taking a wider point of view, how do these models compare not on the basis of performance, but in terms of information that can be derived about human phonotactics? While performance is an important indicator to the principles that might govern human phonotactic judgments, it is important to note the shortcomings of some of these models. First, the use of negative evidence to train the SVM and RNN are clearly not analagous to the human learner. Secondly, while the MaxEnt model chooses explicit constraints to optimize, the neural net architecture is exceedingly difficult to interpret. These are both concerns that should be taken into account along with model performance.

5.2 Training data structure

Though the RNN model does correlate more highly with the Scholes data when trained on the type frequency training data, the maximum entropy model does not change much in its performance. This also creates a problem for the maximum entropy grammar, which, though the model trained with type frequency training data does have a higher correlation with or without the tuning parameter, the discrepancy is incredibly small and will undoubtedly vary with other datasets. I find this challenges the ability of the maximum entropy grammar to accurately be a model of gradient grammaticality. Though the correlation value might be high, looking at the input (in the frequency-equalized case) and the model output (without the tuning parameter), it is not clear that gradient grammaticality is learned or expressed by the model at all. In fact, this model could be used as a binary classifier by drawing a decision boundary through the model outputs, such that if the Maxent value is greater than 0.5 the onset is predicted as grammatical, and if it is lower than 0.5 it is predicted as ungrammatical. This would be analogous to the classification done by the RNN model.

This also complicates the claims of Pierrehumbert (1993) about the relationship between frequency structure in the lexicon and phonotactic acceptability. If this claim is true, models should perform much better when trained with a type frequency structure. The RNN results do not directly contradict the predictions of this claim. Though it competently learns to classify gross status of onsets without frequency structure, which suggests that a model can learn whether an onset is attested or unattested without frequency structure, it still has a higher correlation with the Scholes participant responses when trained with the type frequency data. On the other hand, the Maxent model maintains a nearly identical correlation with the Scholes data regardless of which dataset it is trained on, specifically when untuned scores are reported.

5.3 Experiment data used for evaluation

One concern of current work is the lack of judgment data to use as evaluation for computational models. The Scholes dataset is quite small and could be improved upon greatly. In future work, it will be necessary to run a judgment experiment designed as evaluation data for these models. Evaluating these models on further experimental data will either strengthen or weaken their merit. A future experiment could use a Likert scale rating task to capture gradience in each participant's individual judgments, in which case a correlation metric could assess the likeness of each participant's responses to the group to ensure that intermediate judgments are produced across speakers consistently. This would more accurately represent the phenomenon of gradient acceptability than the Scholes data since the gradience would be represented as an intra-speaker measure, and not an average over yes/no responses.

CHAPTER 6

CONCLUSION

In conclusion, the three main findings are the following: first, for the models used in this thesis, it was found that the maximum entropy model correlates best with the acceptability judgments from the Scholes data, followed by the RNN model and SVM model. Second, one of the models with the capability to provide gradient scores for the onset predictions provide scores on a continuum, but predict onsets as either highly acceptable or highly unacceptable. Lastly, using training data with equalized frequency structure (where the number of onsets in the training is equal for each unique onset) does not significantly impact the maximum entropy model performance, slightly worsens RNN performance on a withheld test set, and significantly hinders RNN performance in correlating with Scholes' judgment data.

I claim that this shows that a model providing a high correlation value with human judgment data is not enough to make a convincing case for gradient phonotactic grammaticality. The motivation for a probabilistic account is the nature of gradient acceptability judgments found by Albright and Hayes (2003) and Albright (2007). Regardless of the correlation value or metric used for evaluation, if the model does not output a range of gradient judgments which not only correlate well with the data, but can accurately predict the intermediate judgments, it has failed to capture the gradience that prompted the modeling to begin with.

More merit should be given to categorical approaches when modeling phonotactics, and nuance around the interaction between the probabilistic and categorical pieces should be carefully explained and tested. Phonotactics is only one of many areas of linguistics where researchers are increasingly interested in testing the potential of deep learning methods as a way to learn more about linguistic knowledge. However, a number of assumptions make it difficult to not be mislead by the results of the models, as they can only offer analogies and can be difficult to interpret.

Both the SVM and the RNN underperform compared to previous work. I think this

could be due to the imbalance of the two-class training. These are both discriminatory models requiring positive and negative data to learn, but generating negative data creates an imbalanced dataset resulting in both models overfitting to the negative data, with a high rate of false negatives.

The phonotactic learner provides an advantage over neural models in providing specifically generated constraints, and correlates well with the Scholes data. However, the model fails to capture any notion of intermediate judgments or gradience without being tuned to the test data that it is being validated upon. Moreover, it does not seem to rely on any frequency structure in the data to perform well.

Correlation values can mislead interpretation if not presented alongside a visual plot of the data. Anscombe (1973) showed four data distributions now known as *Anscombe's Quartet*, which all have very different distributions but extremely similar correlation values. For this reason, results of phonotactics models should not be boiled down to a descriptive statistic like Pearson's r, as this is not a full description of the result. Though the phonotactic learner itself is probabilistic in nature, I believe it cannot be claimed as a model of gradient grammaticality or acceptability for this reason. Moreover, it does not seem to rely on any frequency structure in the data to perform well.

If the goal, above all else, is to find a model to correlate with the data in a way that reproduces the intermediate judgments, the best possibility might lie in a RNN language model (Mikolov et al. 2010; Mayer and Nelson 2019). Mayer and Nelson (2019) do train a model that correlates better with judgment data than the maxent phonotactic learner.

However, I believe the results of this thesis show that recent modeling approaches have not investigated fully the different assumptions these models are relying on. It is not clear to me that the maxent phonotactic learner is evidence for gradient grammaticality; and though a neural network might be performing marginally better with more gradient output, neural nets are notoriously difficult to interpret. There are no constraints that are generated, and everything the model has learned is contained in a "black box" of weights and biases inside

the network.

The ability of a neural net to correlate with human judgments is exciting and should be pursued further, but at this point I do not think anything can be said about how the neural network performance informs us about human grammaticality knowledge. Though I agree with Pater (2019), Mayer and Nelson (2019), and Mirea and Bicknell (2019) that the integration of neural network modeling and linguistics is a promising and thrilling future for the field, I believe extra care must be taken to continually be sure that when these models are learning, that we the researchers are learning something as well.

BIBLIOGRAPHY

BIBLIOGRAPHY

Albright, Adam (2009). "Feature-based generalization as a source of gradient acceptability." In: *Phonology*.

— (2007). "Natural classes are not enough: Biased generalization in novel onset clusters." In: *15th Manchester Phonology Meeting, Manchester, UK*.

Albright, Adam and Bruce Hayes (2003). "Rules vs. analogy in English past tenses: A computational/experimental study." In: *Cognition*.

Anscombe, Francis J (1973). "Graphs in statistical analysis". In: *The american statistician* 27.1, pp. 17–21.

Armstrong, Sharon Lee, Lila Gleitman, and Henry Gleitman (June 1983). "What Some Concepts Might Not Be". In: *Cognition* 13, pp. 263–308. DOI: 10.1016/0010-0277(83)90012-4.

Bailey, Todd M and Ulrike Hahn (2001). "Determinants of wordlikeness: Phonotactics or lexical neighborhoods?" In: *Journal of Memory and Language* 44.4, pp. 568–591.

Chomsky, Noam (1965). *Aspects of the Theory of Syntax*. 50th ed. The MIT Press. ISBN: 9780262527408. URL: http://www.jstor.org/stable/j.ctt17kk81z.

Chomsky, Noam and Morris Halle (1965). "Some controversial questions in phonological theory." In: *Journal of Linguistics*.

— (1968). *The Sound Patterns of English*. New York, Harper and Row.

Coltheart, M. (1977). "Access to the internal lexicon". In: *The psychology of reading*. URL: https://ci.nii.ac.jp/naid/10018074200/en/.

Cover, Thomas M. and Joy A. Thomas (1991). *Elements of information theory*. New York: Wiley.

Daland, Robert et al. (Aug. 2011). "Explaining sonority projection effects". In: *Phonology* 28. DOI: 10.1017/S0952675711000145.

Della Pietra, Stephen, Vincent Della Pietra, and John Lafferty (1997). "Inducing Features of Random Fields". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Dupoux, Emmanuel et al. (Feb. 2004). "Epenthetic Vowels in Japanese: a Perceptual Illusion?" In: *Journal of Experimental Psychology Human Perception & Performance* 25. DOI: 10.1037//0096-1523.25.6.1568.

Durvasula, Karthik et al. (Feb. 2018). "Phonology modulates the illusory vowels in perceptual illusions: Evidence from Mandarin and English". In: *Laboratory Phonology* 9. DOI: 10.5334/labphon.57.

Elman, Jeffrey L. (1990). "Finding structure in time." In: *Cognative Science*.

Goldwater, Sharon and Mark Johnson (2003). "Learning OT constraint rankings using a maximum entropy model". In: *Proceedings of the Stockholm workshop on variation within Optimality Theory*. Vol. 111120.

Gorman, Kyle (2013). "Generative Phonotactics". PhD thesis. University of Pennsylvania.

Halle, Morris (1962). "Phonology in generative grammar." In: *Word*.

— (1959). *The Sound Pattern of Russian*. The Hague: Mouton.

Hay, Jennifer, Janet Pierrehumbert, and Mary Beckman (2004). "Speech Perception, Well-formedness, and the Statistics of the Lexicon". In: *Papers in Laboratory Phonology* VI, pp. 58–74.

Hayes, Bruce and Colin Wilson (2008). "A Maximum Entropy Model of Phonotactics and Phonotactic Learning". In: *Linguistic Inquiry*.

Jaynes, Edwin T. (1983). *Papers on probability, statistics, and statistical physics*. USA: Kluwer Boston. ISBN: 9027714487.

Jurafsky, Daniel and James H. Martin (2009). *Speech and Language Processing (2nd Edition)*. USA: Prentice-Hall, Inc. ISBN: 0131873210.

Kabak, Baris and William Idsardi (Feb. 2007). "Perceptual Distortions in the Adaptation of English Consonant Clusters: Syllable Structure or Consonantal Contact Constraints?" In: *Language and speech* 50, pp. 23–52. DOI: 10.1177/00238309070500010201.

Mayer, Connor and Max Nelson (Oct. 2019). "Phonotactic learning with neural language models". In:

McKinney, Wes (2010). "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman, pp. 51–56.

Mikolov, Tomas et al. (Jan. 2010). "Recurrent neural network based language model". In: vol. 2, pp. 1045–1048.

Mirea, Nicole and Klinton Bicknell (2019). "Using LSTMs to Assess the Obligatoriness of Phonological Distinctive Features for Phonotactic Learning". In: *ACL*.

New, Boris and Christophe Pallier (2009). *SubtlexUS - Lexique*. URL: lexique.org/?page_id=241 (visited on 04/01/2020).

Paszke, Adam et al. (2017). "Automatic differentiation in PyTorch". In: *NIPS-W*.

Pater, Joe (2019). "Generative linguistics and neural networks at 60: foundation, friction, and fusion." In: *Language*.

Pedregosa, F. et al. (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Pierrehumbert, Janet (1993). "Prosody, Intonation, and Speech Technology". In: ed. by M. Bates and R. Weischedel. Cambridge, UK: Cambridge University Press, pp. 257–282.

Python Software Foundation (n.d.). *Python Language Reference*. Version 3.6.6. URL: https://python.org.

Sarver, Isaac (May 2020). *Phonotactics Models*. DOI: 10.17605/OSF.IO/F76ZN. URL: osf.io/f76zn.

Scholes, Robert J. (1966). *Phonotactic Grammaticality*. Mouton.

Schütze, Carson (Mar. 2011). "Linguistic Evidence and Grammatical Theory". In: *Wiley Interdisciplinary Reviews: Cognitive Science* 2, pp. 206–221. DOI: 10.1002/wcs.102.

Shademan, Shabnam (2006). "Is Phonotactic Knowledge Grammatical Knowledge ?" In:

Vetterling, William T. et al. (Nov. 1992). *Numerical Recipes Example Book C (The Art of Scientific Computing)*. 2nd. Cambridge University Press. ISBN: 0521437202.

Weide, Robert L. (1998). *The CMU Pronouncing Dictionary*. http://www.speech.cs.cmu.edu/cgi-bin/cmudict. Accessed: 2019-09-14.