A Learning-Based Account of Phonological Tiers

Caleb Belth

Morphophonological alternations often involve dependencies between adjacent segments. Despite the apparent distance between relevant segments in the alternations that arise in consonant and vowel harmony, these dependencies can usually be viewed as adjacent on a tier representation. However, the tier needed to render dependencies adjacent varies cross-linguistically, and the abstract nature of tier representations in comparison to flat, string-like representations has led phonologists to seek justification for their use in phonological theory. In this paper, we propose a learning-based account of tier-like representations. We argue that humans show a proclivity for tracking adjacent items, and propose a simple learning algorithm that incorporates this proclivity by tracking only adjacent dependencies. The model changes representations in response to being unable to predict the surface form of alternating segments—a decision governed by the Tolerance Principle, which allows for learning despite the sparsity and exceptions inevitable in naturalistic data. Tier-like representations naturally emerge from this learning procedure, and, when trained on small amounts of natural language data, the model achieves high accuracy generalizing to held-out test words, while flexibly handling cross-linguistic complexities like neutral segments and blockers. The model also makes precise predictions about human generalization behavior, and these are consistently borne out in artificial language experiments.

1 Introduction

Phonological theory, and linguistic theory more broadly, often attempts to interpret apparently long-distance dependencies as being adjacent on some representation. For many morphophonological alternations, dependencies are already adjacent in a straight-forward, flat representation. Such is the case for the English plural morpheme, which alternates between [-z] and [-s], matching the stem-final segment's voicing, and separated by [ə] if the stem-final segment is a sibilant, as in (1).

(1) [dɑg-z]
[kæt-s]
[hɔrs-əz]

However, vowel and consonant harmony (Rose and Walker, 2004; Van der Hulst, 2016) and dissimilation (Bennett, 2013) often involve dependencies between segments that are arbitrarily distant in a flat representation. For instance, the vowels of Turkish suffixes harmonize with the preceding vowel across intervening consonants. The affix vowels in (2) alternate between back {ɑ, ɯ} and front {e, i} to match the backness of the preceding vowel.

(2)	[dɑl-lɑr-ɯn]	branch-PL-GEN	(Kabak, 2011, p. 3)
	[jer-ler-in]	place-PL-GEN	(Kabak, 2011, p. 3)
	[ip-ler-in]	rope-PL-GEN	(Nevins, 2010, p. 28)

This harmony process can be viewed as a local spreading process across a phonological *tier* containing only the vowels (Goldsmith, 1976; Clements, 1976, 1980). A similar analysis is possible for other alternations. For instance, Aari exhibits sibilant harmony, as exemplified in (3; McMullin 2016, p. 21; Hayward 1990), where underlying /s/ (3a) surfaces as [ʃ] when preceded by a [−ant] sibilant at any distance (3b). This can be viewed as a local process on a sibilant tier.

(3)	a.	/baʔ-s-e/	→	[baʔse]	‘he brought’
	b.	/ʔuʃ-s-it/	→	[ʔuʃʃit]	‘I cooked’
		/ʒaʔ-s-it/	→	[ʒaʔʃit]	‘I arrived’
		/ʃed-er-s-it/	→	[ʃederʃit]	‘I was seen’

However, the tier needed to render dependencies local varies extensively cross-linguistically. For instance, to view Finnish’s vowel harmony as local requires excluding the vowels {i, e} from the tier, as they neither participate in nor block harmony: In (4) (Ringen and Heinämäki, 1999, p. 305), the essive case vowel alternates between back [ɑ] and front [æ] depending on the final harmonizing vowel of the stem (4a), but passes over both consonants and neutral vowels—like [−back] [i] in (4b).

(4)	a.	[pøytæ-næ]	table-ESS
		[pouta-na]	fine weather-ESS
	b.	[koti-na]	home-ESS

In Latin, default /l/ (5a) dissimilates to [r] when preceded by /l/ across varying distances (5b), but the dissimilation is blocked by an intervening /r/ (5c) or an intervening [−cor] consonant (5d) (examples from Cser 2010, organized following McMullin 2016).

(5)	a.	*nav-alis*	'naval'
	b.	*popul-aris*	'popular'
		lun-aris	'lunar'
	c.	*flor-alis*	'floral'
	d.	*pluvi-alis*	'rainy'
		leg-alis	'legal'

Consequently, in addition to [l] and [r], the relevant tier must also contain the [−cor] consonants to block the dissimilation (McMullin, 2016; Burness *et al.*, 2021).

Because of the cross-linguistic variation demonstrated by these examples and others (McMullin and Hansson, 2016; Burness *et al.*, 2021), together with the abstractness of tier representations in comparison to flat representations, phonologists have sought justification for their use in phonological theory. Hayes and Wilson (2008) proposed an *inductive baseline* argument by demonstrating that their phonotactic learner fails to learn non-local generalizations unless it is provided a relevant tier projection *a priori*. Hayes and Wilson viewed these results as learning-based evidence in favor of the use of tiers in phonological theory. Formal-language-theoretic results demonstrate that restricting a learner's hypothesis space to tier-based generalizations allows for proving strong, theoretical learning results (Heinz *et al.*, 2011; McMullin, 2016; Burness and McMullin, 2019, 2021), which provides theoretical support for Hayes and Wilson (2008)'s empirical results. Goldsmith and Riggle (2012) demonstrated that tier representations allow for better compression of linguistic data, and viewed this as statistical justification for the use of tier representations.

In this paper, we seek to approach this important question from the opposite direction, by building a computational model bottom-up (§ 2) from an independently-motivated psychological mechanism—namely, results from sequence learning experiments, which forcefully demonstrated that learners more readily track dependencies between adjacent items (Saffran *et al.*, 1996, 1997; Aslin *et al.*, 1998) than non-adjacent items (Santelmann and Jusczyk, 1998; Gómez, 2002; Newport and Aslin, 2004; Gómez and Maye, 2005) (see § 2.1). We demonstrate that attention being focused on adjacent dependencies motivates the removal of any segments preventing prediction of the alternating segment's surface form, effectively creating a new representation, which can be interpreted as a tier. This process mirrors the iterative erasure mechanism proposed by Jardine and Heinz (2016) and Burness and McMullin (2019, 2021) as a natural consequence of the formal properties of Tier-Strictly-2-Local languages and functions. This convergent development from different perspectives supports an iterative change of representation as a possible mechanism involved in phonological learning. Moreover, our model uses the Tolerance Principle (Yang, 2016) to determine when a change of representation is needed, which improves the model's robustness against the exceptions and sparsity of naturalistic data. In § 3, we discuss prior models for learning non-local phonological dependencies, and in § 4 compare our proposed model—and prior models—to human behavior on multiple prior artificial language experiments. Our model is the only one to match human behavior in all cases. In § 5, we evaluate our model at learning natural language, non-local alternations. Our model effectively learns Turkish vowel harmony, Finnish vowel harmony, and Latin liquid dissimilation, with substantially greater test accuracy than prior models. These results demonstrate that the model can handle parasitic harmony (Turkish secondary rounding harmony), neutral segments (Finnish vowel harmony), and blocking segments (Latin liquid dissimilation). The model achieves these results on datasets characteristic of the size of children's vocabularies at the time they appear to begin extending harmony to novel words (Altan, 2009). Because of the model's basis in an independently-motivated psychological mechanism, we argue that these results provide a possible learning-based account of tier representations.

2 Model Description

Our model, which we call D2L for *Distant To Local*, is based on experimental results in sequence learning, which have demonstrated that learners track adjacent dependencies more readily than non-adjacent dependencies. We review these results in § 2.1.

The input to D2L is a set, V, of (UR, SR) input-output pairs. We follow the commonly-adopted position that learners only construct abstract URs when doing so is necessary to account for surface alternations (Kiparsky, 1968; Peperkamp *et al.*, 2006; Ringe and Eska, 2013; Tesar, 2013; O'Hara, 2017; Richter, 2018, 2021; Belth, 2023b,c), and treat segments that alternate on the surface as underlyingly underspecified. This is consistent with Nevins

(2010)'s treatment of harmony as an alternating segment searching for a value, and Richter (2018, 2021); Belth (2023c,b,b) provide accounts of how surface alternation can be used to construct such URs. Future work will focus on the important problem of jointly learning URs and processes mapping between URs and SRs (Cotterell *et al.*, 2015; Hua *et al.*, 2020; Ellis *et al.*, 2022).

D2L attempts to form a generalization to predict the surfacing of the alternating segments, *A*, in terms of segments adjacent to them. If this fails, D2L deletes the adjacent segments that failed to account for the alternation and tries again. This process iterates until an adequate generalization is discovered or until no more segments can be deleted. In § 2.2, we describe the structure of D2L's generalizations: how they project a tier and how the surface form of alternating segments is predicted from adjacent segments. Then in § 2.3, we describe how D2L automatically constructs such generalizations.

2.1 Experimental Studies of Sequence Learning

Studies of statistical sequence learning have studied whether learners can segment a continuous speech stream into discrete units based on statistical information (Saffran *et al.*, 1996, 1997; Aslin *et al.*, 1998). These studies hypothesized that infants could track *transitional probabilities*, which capture the strength of adjacent dependencies in sequence data, and use them as a cue to segment speech. These studies found infants as young as 8-months to be sensitive to dependencies between adjacent syllables (Saffran *et al.*, 1996, 1997; Aslin *et al.*, 1998). The ability to track adjacent dependencies has also been attested between morphemes (Santelmann and Jusczyk, 1998), non-linguistic tones (Saffran *et al.*, 1999), and visual shapes (Fiser and Aslin, 2002). The diversity of types of elements for which this ability has been observed suggests that the ability to track adjacent dependencies may be neither limited to a particular kind of linguistic structure nor even the domain of language.

Statistical learning studies were also directed towards non-adjacent dependencies. While the ability to track adjacent dependencies has been widely attested even for infants as young as 8 months old, Santelmann and Jusczyk (1998) found no evidence of learners tracking non-adjacent dependencies even at 15-months-old.

The ability to track non-adjacent dependencies does eventually emerge. Adults show a sensitivity to dependencies between non-adjacent phonological segments (Newport and Aslin, 2004), and 18-month-old children show sensitivity to dependencies between both non-adjacent morphemes (Santelmann and Jusczyk, 1998) and words (Gómez, 2002). Gómez and Maye (2005) attempted to map the developmental trajectory of this ability to track non-

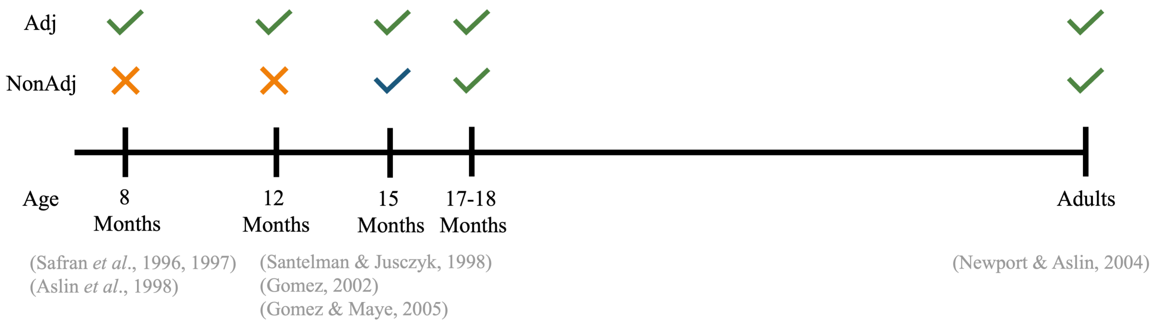

Figure 1: The asymmetrical developmental trajectory of learners' ability to track adjacent and non-adjacent dependencies. Infants as young as 8mo old show evidence of being able to track adjacent dependencies; this ability persists into adulthood. In contrast, learners do not show evidence of tracking non-adjacent dependencies until around 15mo.

adjacent dependencies, and found that it grew gradually with age. At 12 months, infants showed no evidence of tracking non-adjacent dependencies, but they began to do so by 15 months, and showed further advancement at 17 months.

Thus, the developmental trajectory of adjacent and non-adjacent dependencies, shown in Fig. 1, is asymmetrical. Infants as young as 8-months-old show evidence of tracking adjacent dependencies, and this ability persists into adulthood, but infants only begin to show evidence of tracking non-adjacent dependencies at around 15 months.

Even as sensitivity to non-adjacent dependencies develops, learners still more readily track local dependencies. Gómez (2002) found that 18mo-olds could track non-adjacent dependencies, but that they only did so when adjacent dependencies were unavailable. Gómez and Maye (2005) replicated these results with 17mo-olds and described the situation like this (p. 199): 'It is as if learners are attracted by adjacent probabilities long past the point that such structure is useful.' Indeed, artificial language experiments have repeatedly demonstrated that learners more easily learn local phonological processes than non-local ones (Baer-Henney and van de Vijver, 2012), and, when multiple possible phonological generalizations are consistent with exposure data, learners systematically construct the most local generalization (Finley, 2011; White *et al.*, 2018; McMullin and Hansson, 2019).

The evidence that infants track adjacent dependencies across a range of items not limited to the linguistic domain, that the ability to track non-adjacent dependencies appears to emerge later in development than that for adjacent dependencies, and that learners only show evidence of tracking non-adjacent dependencies as a last resort when adjacent dependencies fail them, strongly supports the conclusion that human's exhibit a proclivity for adjacency. This proclivity constitutes an independent foundation for our proposed model.

2.2 The Structure of Generalizations

D2L constructs a local rule, from which the surface form of alternating segments (the rule's target) can be predicted from left-adjacent or right-adjacent segments after a (possibly empty) set of segments has been deleted. Consequently, we characterize a generalization as the composition of a local rule, r_{adj}, with a tier projection, proj(x, T), which preserves only the segments in input sequence $x = x_i, \ldots, x_n$ that are contained in the tier $T \subseteq \Sigma$ (6).

(6) $g(x) = r_{\text{adj}} \circ \text{proj}(x, T)$

Clearly, the proj$(\cdot, T)$ can interchangeably be viewed as creating a new sequence in which only segments in T are preserved, or as creating a new sequence in which all segments not in T are deleted. Consequently, it will sometimes be useful to refer to the complement of T as the **deletion set** $D \triangleq \Sigma \setminus T$. Thus, tier projection is an *erasing function* (Heinz *et al.*, 2011), which creates a new sequence $t_1, \ldots, t_m$, where every segment $x_i \notin T$ is deleted. The tier-segments are annotated with their indices in the original sequence so that the results of the rule application can be written to the correct position in the output sequence. For instance, a [+sib][1] tier projection is shown in (7) for a hypothetical input. We use '<·>' to denote a tier projection. The superscripts denote the annotation of each sibilant's position in the input sequence.

(7) proj(/ʃoku-s-is/, [+sib]) = <$\text{ʃ}^{(1)}\text{s}^{(5)}\text{s}^{(7)}$>

Once the tier is projected, the local rule r_{adj} applies over it. This r_{adj} has the structure of an SPE rule (Chomsky and Halle, 1968) with either a left (8a) or a right context (8b).

(8) a. $A \longrightarrow B \,/\, C \,\_\_$
b. $A \longrightarrow B \,/\, \_\_ \, C$

In this paper, $A \longrightarrow B$ can either be $\text{Agree}(A, F)$, which sets A's features $f \in F$ to match those in C, or $\text{Disagree}(A, F)$, which sets them to the opposite of C's. This follows Nevins (2010) in drawing analogy to syntactic Agree (Chomsky, 2001b,a). We discuss this point further in the appendix (§ A.2). This brings the rule schemas to those in (9).

[1] For ease of exposition, we use [+sib] to refer to a set of sibilants in a language's segment inventory, not as a commitment a universal feature [+sib] with that extension. Our experiments do not use a [+sib] feature and can only reference sibilants via a [stri] feature.

(9) a. $\textsc{Agree}(A, F)\ /\ C\ \_\_$
$\textsc{Agree}(A, F)\ /\ \_\_\ C$

b. $\textsc{Disagree}(A, F)\ /\ C\ \_\_$
$\textsc{Disagree}(A, F)\ /\ \_\_\ C$

An example sibilant harmony rule is (10).

(10) $g(x) = \textsc{Agree}([\text{+sib}], \{\text{ant}\}) / [\text{+sib}]\_\_ \circ \text{proj}(x, [\text{+sib}])$

Because harmony patterns tend to spread (Nevins, 2010; Burness *et al.*, 2021), the rules apply iteratively.[2] If the rule has a left-context, it applies left-to-right; otherwise it applies right-to-left.

The output sequence is generated by copying every non-tier element to the output directly, and copying every tier element to the output position corresponding to its annotated index. Example (11) shows the sibilant rule (10) applying to a hypothetical input—underlines show the rule applications.

(11) /ʃoku-s-is/ → <ʃ$^{(1)}$s$^{(5)}$s$^{(7)}$> → <$\underline{\text{ʃ}^{(1)}\text{ʃ}^{(5)}}s^{(7)}$> → <ʃ$^{(1)}\underline{\text{ʃ}^{(5)}\text{ʃ}^{(7)}}$> → [ʃokuʃiʃ]

The restriction of operations to Agree and Disagree and the context to either a single position to the left or right is to keep the model and its discussion succinct. To handle epenthesis, deletion, and contexts larger than a single segment (e.g., intervocalic voicing), D2L could be placed in a broader phonological learning framework like Belth (2023a)'s.

Since the rules are applied iteratively and invoke a tier-local left or right context, they relate to Output Tier-based Strictly 2-Local functions (Burness and McMullin, 2019; Burness *et al.*, 2021). Thus, the algorithm we describe in § 2.3, which is grounded in a sensitivity to adjacent dependencies (§ 2.1), attempts to provide bottom-up support for that class of functions. The appendix includes discussions of connections with search-and-copy accounts of vowel harmony (§ A.1), and possible connections to syntactic Agree (§ A.2).

2.2.1 Examples and Expressivity

These tier-based generalizations can express a wide range of behaviors, including *transparency, parasitic harmony*, and *blocking*.

[2] To learn non-iterative processes, the model can straight-forwardly be extended by applying rules simultaneously instead.

Transparent segments do not participate in an alternation. For example, as described in § 1, the Finnish vowels {i, e} are transparent in vowel harmony (4). This can be expressed by excluding the neutral vowels from the tier $T = [-\text{cons}] \setminus \{\text{i, e}\}$. Since all tier vowels harmonize, and r_{adj} applies over the tier projection, the context can be $C = [-\text{cons}]$. This is stated in (12), where $A = [?\text{back}]$ targets vowels unspecified for [back], causing them to harmonize with a vowel to the left on the tier, which includes all vowels except {i, e}.

(12) $\textsc{Agree}([?\text{back}], \{\text{back}\}) / [-\text{cons}]\ \_\_ \circ \text{proj}(\cdot, [-\text{cons}] \setminus \{\text{i, e}\})$

For example, if the ESS suffix vowel in [ko̲tinɑ̲] ‘home’-ESS from (4b) is underlyingly underspecified for [back], rule (12) would derive the surface form as in (13). Here /A/ is the /−round,+low,?back/ vowel, which alternates between [+back] [ɑ] and [−back] [æ].

(13) $/\text{koti-nA}/ \rightarrow <\text{o}^{(2)}\text{A}^{(6)}> \rightarrow <\underline{\text{o}^{(2)}\text{ɑ}^{(6)}}> \rightarrow [\text{k}\underline{\text{o}}\text{tin}\underline{\text{ɑ}}]$

In *parasitic harmony*, segments only harmonize with respect to some feature when they agree in another feature. In Kachin Khakass, only [+high] {i, ɯ, y, u} suffixal vowels undergo rounding harmony, and only with [+high] stem vowels (14a). They fail to harmonize with [−high] {e, ø, a, o} stem vowels (14b) and [−high] suffixal vowels never harmonize (14c). Data from Korn (1969); Burness *et al.* (2021, p. 18).

(14)	a.	[ky̲n-ny̲]	‘day-ACC’
		[ku̲ʃ-tu̲n]	‘of the bird’
	b.	[ok-tɯn]	‘of the arrow’
	c.	[kyn-ge]	‘to the day’
		[pol-za]	‘if he is’

This can be captured by including only [+high] vowels on the tier: $T = [+\text{high}, -\text{cons}]$ as in (15). The context can again be [−cons] since the tier already excludes [−high] vowels.

(15) $\textsc{Agree}([?\text{round}], \{\text{round}\}) / [-\text{cons}]\ \_\_ \circ \text{proj}(\cdot, [+\text{high}, -\text{cons}])$

In some cases, such as Turkish secondary rounding harmony (see § 5.1), [+high] vowels undergo rounding harmony with vowels of any height. This can be expressed by including all vowels on the tier, while excluding [−high] vowels from the target (16).[3]

(16) $\textsc{Agree}([-\text{high}, ?\text{round}], \{\text{round}\}) / [-\text{cons}]\ \_\_ \circ \text{proj}(\cdot, [-\text{cons}])$

[3] Alternatively, if only alternating vowels are underspecified, [−high] vowels would be fully specified underlyingly and thus excluded automatically by the [?round] condition.

In some cases, harmony or dissimilation is *blocked* by some segments. For example, in Khalkha Mongolian (Nevins, 2010, p. 137) (17) the rounding harmony in (17a) is blocked by the [+round] vowels {u, ʊ} (17b).

(17)	a.	[to̲r-o̲ːd]	‘be.born-PERF’
		[ɔ̲r-ɔ̲ːd]	‘enter-PERF’
	b.	[tor-uːl-eːd]	‘be.born-CAUS-PERF’
		[ɔr-ʊːl-aːd]	‘enter-CAUS-PERF’

Critically, the PERF affix vowels {e, a} in (17b) are [−round], implying that they do not harmonize opaquely with the [+round] {u, ʊ} blockers. This blocking can be expressed by including [+round] on the tier, but excluding [+round] from the context (18).

(18) $\textsc{Agree}([?\text{round}], \{\text{round}\})/[-\text{round}]\ \_\_ \circ \text{proj}(\cdot, [-\text{cons}])$

This must be combined with a default [−round] value to account for [−round] vowels surfacing in (17b) when harmony is blocked. We discuss discovering defaults in § 2.3.3.

2.3 Learning

D2L follows the steps in (19). The cases of assimilation and dissimilation are symmetrical. For clarity, we describe D2L in the context of assimilation, and discuss in § 2.3.4 how D2L automatically infers whether the alternation is assimilatory or dissimilatory.

```
(19)  **Input:** (UR, SR) pairs $V$ and a set of segments $A$ that alternate on features $F$
      1. Initialize tier $T = \Sigma$ (equivalently deletion set $D = \emptyset$)
      2. **While** $T \neq \emptyset$ **do**
      3. – $g_l = \textsc{Agree}(A, F)/C_l\_\_ \circ \text{proj}(\cdot, T)$
      4. – $g_r = \textsc{Agree}(A, F)/\_\_C_r \circ \text{proj}(\cdot, T)$
      5. – $g = \arg\max_{g \in \{g_l, g_r\}} \text{acc}(g, V)$                         (§ 2.3.1)
      6. – **If** $\text{sat}(g, V)$ **then**        ▹ Checks if rule is sufficiently accurate   (§ 2.3.1)
      7. — **Return** $g$
      8. – Remove from $T$ segments adj. to $A$ on $T$ that cannot account for the alternation
                                                                                     (§ 2.3.2)
```

The input to D2L is a set of input-output (UR, SR) pairs, such as (20a), where we use /S/ for alternating sibilants.[4] The alternating segments, A, and what features they alternate

[4] The default form of /S/ is [s]; we discuss how D2L discovers this in § 2.3.3.

on, F, are directly computed from discrepancies between these inputs and outputs (20b). In (20a), since sometimes /S/ ⟶ [ʃ] and sometimes /S/ ⟶ [s], ‘S’ $\in A$ and ‘ant’ $\in F$. In our presentation of D2L, we treat alternating segments as underlyingly underspecified (e.g. /S/). We will use (20) as a toy example throughout our presentation of D2L. The (UR, SR) pairs are (20a) and the alternating segments and features are (20b).

(20) a. /ʃoku-S-iS/ ⟶ [ʃokuʃiʃ]
/apʃa-S/ ⟶ [apʃaʃ]
/ʃun-iS/ ⟶ [ʃuniʃ]
/soki-S/ ⟶ [sokis]
/sigo-S-iS/ ⟶ [sigosis]
/ut-S/ ⟶ [uts]

b. $A = \{\text{S}\}$, $F = \{\text{ant}\}$

Initially no segments are deleted, so $T = \Sigma$ (19; step 1). After initialization, D2L enters the while-loop (19; step 2) and constructs left and right rules (19; step 3)-(19; step 4). To do so, the set F is constructed to contain the features that differ across A’s surface realizations, and the sets C_l/C_r are constructed to contain every segment tier-adjacent (on the left for g_l and the right for g_r) to an alternating segment. For the words in (20), the first left and right rules are those in (21) because the segments to the left of /S/ are {u, i, a, o, t} and to the right of /S/ are {i, ⋉}, where ‘⋉’ denotes a right word boundary.

(21) Tier, deletion set, and rules at the first iteration
$T = \Sigma, D = \emptyset$
$g_l = \textsc{Agree}(\{\text{S}\}, \{\text{ant}\})/\{\text{u, i, a, o, t}\}\_\_ \circ \text{proj}(\cdot, \Sigma)$
$g_r = \textsc{Agree}(\{\text{S}\}, \{\text{ant}\})/\_\_\{\text{i}, \ltimes\} \circ \text{proj}(\cdot, \Sigma)$

The accuracy of these rules is then computed (19; step 5) and the more accurate rule is checked to see if it is a sufficiently good generalization (19; step 6)

2.3.1 Computing Accuracy and Rule Quality

The accuracy of a rule g is straight-forwardly defined (22c) as the number of correct predictions made by g over the training instances V (22b), divided by its total number of predictions over the training instances (22a).

(22) a. $n(g, V) \triangleq$ number of g’s predictions over V

b. $c(g,V) \triangleq$ number of g's correct predictions over V

c. $\mathrm{acc}(g,V) \triangleq \frac{c(g,V)}{n(g,V)}$

Since rules are applied iteratively, the number of applications and correct applications are computed iteratively as well. Thus, the sibilant harmony rule (23a), which states that underlying /S/ should agree in anteriority with the preceding sibilant (after projecting a [+sib] tier), has two applications (both correct) over the input (23b). We denote rule applications with underlines.

(23) a. $\text{Agree}(\{\text{S}\},\{\text{ant}\})/[\text{+sib}]\_\_ \circ \text{proj}(x, [\text{+sib}])$

b. /ʃoku-S-iS/ → $<\text{ʃ}^{(1)}\text{S}^{(5)}\text{S}^{(7)}> \rightarrow <\underset{\textbf{(+1)}}{\underline{\text{ʃ}^{(1)}\text{ʃ}^{(5)}}}\text{S}^{(7)}> \rightarrow <\text{ʃ}^{(1)}\underset{\textbf{(+1)}}{\underline{\text{ʃ}^{(5)}\text{ʃ}^{(7)}}}>$ → [ʃokuʃiʃ]

An application is considered incorrect if either (a) it predicts the incorrect surface form, or (b) the target cannot Agree/Disagree with the contextual segment due to the relevant feature being unspecified. For example, consider rule (24a), which states that underlying /S/ should agree in anteriority with the preceding consonant. This rule will make a correct prediction for /apʃa-S/ → [apʃaʃ] in (24b), since the underlying /S/ taking its anteriority from /ʃ/ indeed leads to it correctly surfacing as [ʃ]. However, for /ʃun-iS/ → [ʃuniʃ], /S/ harmonizing with [+ant] [n] will lead to incorrect surface form [+ant] [s] (24c).

(24) a. $\text{Agree}(\{\text{S}\},\{\text{ant}\})/[\text{+cons}]\_\_ \circ \text{proj}(\cdot, [\text{+cons}])$

b. /apʃa-S/ → $<\text{p}^{(2)}\text{ʃ}^{(3)}\text{S}^{(5)}> \rightarrow <\text{p}^{(2)}\underline{\text{ʃ}^{(3)}\text{ʃ}^{(5)}}>$ → [apʃaʃ] ✓

c. /ʃun-iS/ → $<\text{ʃ}^{(1)}\text{n}^{(3)}\text{S}^{(5)}> \rightarrow <\text{ʃ}^{(1)}\underline{\text{n}^{(3)}\text{s}^{(5)}}>$ → [ʃunis] ✗

A rule like (25a), which states that /S/ should agree in anteriority with the preceding vowel (all segments Σ projected), will produce an error on /apʃa-S/ → *[apʃaS] for the latter reason: /S/ cannot take the feature value for [ant] from /a/, assuming vowels are not specified for consonantal features.

(25) a. $\text{Agree}(\{\text{S}\},\{\text{ant}\})/[-\text{cons}]\_\_ \circ \text{proj}(\cdot, \Sigma)$

b. /apʃa-S/ → $<\text{a}^{(1)}\text{p}^{(2)}\text{ʃ}^{(3)}\text{a}^{(4)}\text{S}^{(5)}> \rightarrow <\text{a}^{(1)}\text{p}^{(2)}\text{ʃ}^{(3)}\underline{\text{a}^{(4)}\text{S}^{(5)}}>$ → [apʃaS] ✗

Thus, for vocabulary (20), rule (24a) makes a correct prediction for only the sibilants preceded by a consonant that correctly predicts the surface form for /S/. This is shown in (26), where [+ant] = {s, n, p, t} and [−ant] = {ʃ, k, g} ('*' marks errors).

(26) /ʃoku-S-iS/ → <ʃ$^{(1)}$k$^{(3)}$S$^{(5)}$S$^{(7)}$> → <ʃ$^{(1)}\underline{\text{k}^{(3)}\text{ʃ}^{(5)}}S^{(7)}$> → <ʃ$^{(1)}k^{(3)}\underline{\text{ʃ}^{(5)}\text{ʃ}^{(7)}}$>
→ [ʃokuʃiʃ]
/apʃa-S/ → <p$^{(2)}$ʃ$^{(3)}$S$^{(5)}$> → <p$^{(2)}\underline{\text{ʃ}^{(3)}\text{ʃ}^{(5)}}$> → [apʃaʃ]
/ʃun-iS/ → <ʃ$^{(1)}$n$^{(3)}$S$^{(5)}$> → <ʃ$^{(1)}\underline{\text{n}^{(3)}\text{s}^{(5)}}$> → [ʃuni*s]
/soki-S/ → <s$^{(1)}$k$^{(3)}$S$^{(5)}$> → <s$^{(1)}\underline{\text{k}^{(3)}\text{ʃ}^{(5)}}$> → [soki*ʃ]
/sigo-S-iS/ → <s$^{(1)}$g$^{(3)}$S$^{(5)}$S$^{(7)}$> → <s$^{(1)}\underline{\text{g}^{(3)}\text{ʃ}^{(5)}}S^{(7)}$> → <s$^{(1)}g^{(3)}\underline{\text{ʃ}^{(5)}\text{ʃ}^{(7)}}$>
→ [sigo*ʃi*ʃ]
/ut-S/ → <t$^{(2)}$S$^{(3)}$> → <$\underline{\text{t}^{(2)}\text{s}^{(3)}}$> → [uts]

There are 8 instances of /S/, and (24a) predicted the correct surface form for 4 of these. Thus, $n(g, V) = 8$, $c(g, V) = 4$, and $\text{acc}(g, V) = 4/8 = 1/2$.

The function $\text{sat}(g, V)$ is a boolean function that returns 'True' iff g is satisfactorily accurate over the training data V. In this work, we use the Tolerance Principle (Yang, 2016) as the criterion due to its cognitive basis and prior success in computational modeling, lexical, and experimental studies (Schuler *et al.*, 2016; Yang, 2016; Richter, 2018; Koulaguina and Shi, 2019; Emond and Shi, 2021; Richter, 2021; Belth *et al.*, 2021). The Tolerance Principle hypothesizes that learners accept a linguistic generalization when it is cognitively more efficient to do so. It provides a quantitative, categorical threshold for this tipping point in terms of the generalization's scope and how many exceptions it has (see Yang 2016, ch. 3 for the threshold's derivation). When the threshold is met and a generalization accepted, the exceptions to the generalization can be lexicalized. In the current work, using the Tolerance Principle for evaluating generalizations is achieved—in terms of (22)—via (27).

(27)

$$\text{sat}(g, V) \triangleq n(g, V) - c(g, V) \leq \frac{n(g, V)}{\ln n(g, V)}$$

For the initial rules (21), both g_l and g_r make $n(g_l, V) = n(g_r, V) = 8$ predictions over the vocabulary (20), because there are 8 underlying /S/'s. The left rule in (21) makes only 1 correct prediction: /unt-S/ → [unts]; the remaining 7 predictions err because the underlying /S/'s cannot harmonize with adjacent vowels. The right rule in (21) makes no correct predictions. Thus, $\text{acc}(g_l, V) = 1/8$ and $\text{acc}(g_r, V) = 0/8$. Since the former is more accurate, it is chosen (19; step 5). However, since $8 - 1 > 8/\ln 8$ (i.e., $7 > 3.85$), $\text{sat}(g, V) =$ 'False' at (19; step 6), and the tier must be updated.

2.3.2 Updating the Tier

In order for the rule to apply to them, the alternating segments A must always be preserved on the tier. Moreover, any segment currently tier-adjacent to an alternating segment from which the correct surface form cannot be computed cannot be on the tier if the alternation is to be predictable from adjacent dependencies. Consequently, these unuseful segments, which are present in the context sets C_l and C_r, are added to the deletion set D (28).

(28) $D \cup \{s \in C_l \cup C_r :$ agreeing with s is not possible or yields the wrong surface form$\}$

In our example, $\{\mathrm{u,\ i,\ a,\ o}\} \cup \{\mathrm{i}, \ltimes\}$ are added to D. The segment $\{\mathrm{t}\}$ is not added because agreeing /S/ to /t/ correctly yields [s]. For segments that do not occur tier-adjacent to an alternating segment (e.g., /k/) or yield the correct surface form (e.g., [+ant] /t/ adjacent to an /S/ that surfaces as [+ant] [s]), no conclusion is drawn about whether to keep them on the tier or remove them. Thus, D2L takes the smallest natural class that contains all of D but none of A, and removes this from the tier (29) at (19; step 8).

(29) $T \setminus \arg\min_{\{\text{nat class } N : D \subseteq N \wedge A \cap N = \emptyset\}} |N|$

The arg min ranges over natural classes,[5] each of which we refer to with N. The condition $D \subseteq N$ requires that all segments to be deleted (D) are included in the natural class (N), so that they are excluded from the tier. The condition $A \cap N = \emptyset$ requires that no alternating segments (A) are included in N, so that they are preserved on the tier. The arg min returns the smallest natural class satisfying these conditions ($|N|$ is the size of N).[6]

If no such natural class exists, it removes D verbatim, allowing for idiosyncratic tiers that do not fit neatly into a natural class. In our example, both [+cons] and [+sib] include $A = \{\mathrm{S}\}$ and exclude $D = \{\mathrm{u,\ i,\ a,\ o}, \ltimes\}$, so N in (29) ranges over their complements [−cons] and [−sib]. The class [−cons] is smaller (only the vowels) than [−sib] (both vowels and non-sibilant consonants), so [−cons] is deleted; equivalently T = [+cons]. This yields the tier projections shown in (30c), from which the rules (30d) are derived in the second iteration of the while loop.

[5] For ease of exposition, we consider only natural classes describable with a single feature, but clearly the same process could apply over natural classes requiring more features to specify.

[6] This is similar to the strategy of Minimal Generalization described by Albright and Hayes (2002, 2003), which constructs a natural class for a set of segments by retaining all features shared by those segments, in order to include as few segments outside the set as possible. Our method differs in that we do not allow segments in the target set A to be swallowed by the deletion set D, since doing so would prevent the target segments from being targeted.

(30) Deletion set, tier, tier projections, and rules at the second iteration

a. D = {u, i, a, o, $\ltimes$}

b. T = [+cons]

c. /ʃoku-S-iS/ → <ʃ$^{(1)}$k$^{(3)}$S$^{(5)}$S$^{(7)}$>
/apʃa-S/ → <p$^{(2)}$ʃ$^{(3)}$S$^{(5)}$>
/ʃun-iS/ → <ʃ$^{(1)}$n$^{(3)}$S$^{(5)}$>
/soki-S/ → <s$^{(1)}$k$^{(3)}$S$^{(5)}$>
/sigo-S-iS/ → <s$^{(1)}$g$^{(3)}$S$^{(5)}$S$^{(7)}$>
/ut-S/ → <t$^{(2)}$S$^{(3)}$>

d. g_l = Agree({S}, {ant})/{k, S, ʃ, n, g, t}__ ∘ proj(·, [+cons])
g_r = Agree({S}, {ant})/__{S, $\ltimes$} ∘ proj(·, [+cons])

The sets C_l = {k, S, ʃ, n, g, t} and C_r = {S, $\ltimes$} are computed from the segments to the left and right of the alternating /S/ on the [+cons] tier. The new rules g_l and g_r again apply to all 8 /S/ segments, so $n(g_l, V) = n(g_r, V) = 8$. The right rule g_r makes 0 correct predictions because /S/ cannot harmonize with '$\ltimes$' or a right-adjacent /S/ that also is not specified for anteriority. The left rule makes 4 correct predictions, as shown in (31): 2 on the first word, 1 on the second, and 1 on the last word ('*' marks errors).

(31) /ʃoku-S-iS/ → <ʃ$^{(1)}$k$^{(3)}$S$^{(5)}$S$^{(7)}$> → <ʃ$^{(1)}$$\underline{k^{(3)}ʃ^{(5)}}S^{(7)}$> → <ʃ$^{(1)}k^{(3)}$$\underline{ʃ^{(5)}ʃ^{(7)}}$>
→ [ʃokuʃiʃ]
/apʃa-S/ → <p$^{(2)}$ʃ$^{(3)}$S$^{(5)}$> → <p$^{(2)}$$\underline{ʃ^{(3)}ʃ^{(5)}}$> → [apʃaʃ]
/ʃun-iS/ → <ʃ$^{(1)}$n$^{(3)}$S$^{(5)}$> → <ʃ$^{(1)}$$\underline{n^{(3)}s^{(5)}}$> → [ʃuni*s]
/soki-S/ → <s$^{(1)}$k$^{(3)}$S$^{(5)}$> → <s$^{(1)}$$\underline{k^{(3)}ʃ^{(5)}}$> → [soki*ʃ]
/sigo-S-iS/ → <s$^{(1)}$g$^{(3)}$S$^{(5)}$S$^{(7)}$> → <s$^{(1)}$$\underline{g^{(3)}ʃ^{(5)}}S^{(7)}$> → <s$^{(1)}g^{(3)}$$\underline{ʃ^{(5)}ʃ^{(7)}}$>
→ [sigo*ʃi*ʃ]
/ut-S/ → <t$^{(2)}$S$^{(3)}$> → <$\underline{t^{(2)}s^{(3)}}$> → [uts]

The successes are when the tier-adjacent consonants {k, ʃ, t} match the surface anteriority of /S/, and the errors are when {n, k, g} do not. Since 4 > 8/ln(8), the rule is still not sufficiently accurate, according to the Tolerance Principle threshold (27). Since the segments {n, k, g} led to incorrect predictions, they are added to D, yielding D = {u, i, a, o, n, k, g, $\ltimes$}. The natural class [+cons] no longer separates A = {S} from D = {u, i, a, o, n, k, g, $\ltimes$}, because D now contains vowels and consonants. However, [+sib] does separate A and D, so T = [+sib] is set at (19; step 8), yielding (32).

(32) Deletion set and tier at the third iteration

$D = \{\text{u, i, a, o, n, k, g, ⋉}\}$
$T = [\text{+sib}]$

At the third iteration, the new left rule (33a), which projects a [+sib] tier, correctly predicts all the surface forms (33b) except for /ut-S/, where it fails to apply because there is no stem sibilant (i.e., the rule underextends). When this happens, D2L attempts to infer a default form for the alternating segment, as discussed in the next section (§ 2.3.3). The right rule g_r will have zero accuracy for the same reason as the prior iteration, so the left rule is evaluated under the Tolerance Principle at (19; step 6). Since g_l applies to only the first 7 instances of /S/ (the 8th being handled by the default case, as we discuss next), $n(g_l, V) = 7$. As shown in (33b), the rule predicts the correct surface form in all 7 cases, so $c(g_l, V) = 7$. Since $0 \leq 7/\ln 7$, this rule is accepted and returned[7] (19; step 7).

(33) Successful rule and its predictions at the third iteration

a. $g_l = \text{AGREE}(\{\text{S}\}, \{\text{ant}\})/\{\text{s, ʃ}\}\_\_ \circ \text{proj}(\cdot, [\text{+sib}])$

b. /ʃoku-S-iS/ → $<\text{ʃ}^{(1)}\text{S}^{(5)}\text{S}^{(7)}>$ → $<\underline{\text{ʃ}^{(1)}\text{ʃ}^{(5)}}\text{S}^{(7)}>$ → $<\text{ʃ}^{(1)}\underline{\text{ʃ}^{(5)}\text{ʃ}^{(7)}}>$ → [ʃokuʃiʃ]
/apʃa-S/ → $<\text{ʃ}^{(3)}\text{S}^{(5)}>$ → $<\underline{\text{ʃ}^{(3)}\text{ʃ}^{(5)}}>$ → [apʃaʃ]
/ʃun-iS/ → $<\text{ʃ}^{(1)}\text{S}^{(5)}>$ → $<\underline{\text{ʃ}^{(1)}\text{ʃ}^{(5)}}>$ → [ʃuniʃ]
/soki-S/ → $<\text{s}^{(1)}\text{S}^{(5)}>$ → $<\underline{\text{s}^{(1)}\text{s}^{(5)}}>$ → [sokis]
/sigo-S-iS/ → $<\text{s}^{(1)}\text{S}^{(5)}\text{S}^{(7)}>$ → $<\underline{\text{s}^{(1)}\text{s}^{(5)}}\text{S}^{(7)}>$ → $<\text{s}^{(1)}\underline{\text{s}^{(5)}\text{s}^{(7)}}>$ → [sigosis]
/ut-S/ → $<\text{S}^{(3)}>$ → [ut*S]

2.3.3 Default Values

When a candidate rule underextends, D2L takes the set of alternating segments that the rule does not account for and computes the set of surface realizations of those underextensions. If they do not alternate, the surface form is taken as the default. For instance, the rule (33a) underextends for the affix sibilant in /ut-S/, which surfaces as [s] (33b). Taking [s] as the default form for /S/ works, since there are no underextensions of (33a) where /S/ surfaces as anything else. If there were such underextensions, D2L would reject the candidate rule and continue the while-loop.

As another example, when Finnish stems contain only neutral vowels, alternating affix vowels are usually [−back] by default (Ringen and Heinämäki, 1999). For example, the

[7] Since many processes involve all segments on the tier, once a successful rule is constructed, if the context set C_l/C_r can be set to equal the tier T without changing the accuracy of the rule, then this is done.

essive affixal vowel, which alternated between [+back] [ɑ] and [−back] [æ] in (4) surfaces as [−back] [æ] when the stem contains only the neutral vowel [e] (34).

(34) [velje-næ] road-ESS (Nevins, 2010, p. 76)

A [back] harmony rule like (12), which excludes neutral vowels {i, e} from the tier, will underextend to words like (34) with only neutral vowels. However, because these underextensions consistently surface as [−back] vowels, D2L infers this as the default.

2.3.4 Assimilation vs. Dissimilation

From observing a segment that alternates, a learner will not immediately know whether the alternation is due to assimilation or dissimilation. However, attempting to account for an assimilatory alternation by dissimilating from something in the phonological environment (or vice versa) is highly unlikely to yield a productive generalization. Thus, figuring out whether an alternation is assimilatory or dissimilatory should not present a serious challenge to learning. Consequently, we run the D2L algorithm (19) twice in parallel—one searching for an assimilatory rule and one searching for a dissimilatory rule. When searching for an assimilatory rule, g_l and g_r are constructed with Agree, and when searching for dissimilatory rule, they are constructed with Disagree. All other aspects are identical. In most conceivable cases, only one search will yield a productive generalization, in which case the generalization from the successful search is chosen. In the unlikely case that both searches yield a generalization, the more accurate one is chosen. We never observed this scenario in any of our experiments.

The reason it is necessary to take this approach, instead of expanding (19; step 3)-(19; step 4) to include two more rules (i.e., left and right dissimilatory rules), is that it is not possible to maintain a single deletion set for both assimilation and dissimilation. In assimilation, the segments to delete are those that assimilating with yields the wrong surface form; in dissimilation they are those that dissimilating from yields the wrong surface form. Running D2L twice in parallel allows for maintaining these two, distinct deletion sets.

2.4 Strict Locality as a Special Case

As is well-recognized in the literature, D2L's use of tiers unifies learning both local and non-local alternations (e.g., Heinz *et al.* 2011; Jardine and Heinz 2016; McMullin 2016). Because it starts with an empty deletion set, a strictly-local alternation—one determined by string-adjacency—will be discovered on the first iteration of the algorithm.

3 Prior Models

Most prior models for learning non-local phonological generalizations, with the exception of Burness and McMullin (2019, 2021)'s models, have focused on phonotactics, whereas D2L is focused on alternations. We group prior models into statistical, § 3.1, formal-language-theoretic § 3.2, and neural network § 3.3 models.

3.1 Statistical Models

Hayes and Wilson (2008) demonstrated that a phonotactic learner sensitive to only fixed-length sequences failed to learn constraints for long-distance patterns. But, if provided a projection of the data onto a relevant tier, the model was able to learn relevant phonotactic constraints on that tier. However, the model did not learn what tier to project. Gouskova and Gallagher (2020) extended the Hayes and Wilson (2008) model to automatically learn projections. The authors observed that many non-local dependencies, despite being arbitrarily far away in principle, often occur within a window of three segments (i.e., a trigram). Their model uses Hayes and Wilson (2008) to extract baseline phonotactic constraints. Some of these are trigram constraints of the form *X[]Y, where X and Y are sets of segments, and [] allows any segment to intervene. Gouskova and Gallagher (2020)'s model then uses these trigram constraints to project a tier, over which additional constraints are learned. In this respect, GG is like D2L, which also first tracks string-local dependencies. But GG tracks non-adjacent X's and Y's, and learns phonotactics, not alternations. GG preserves both X and Y on the tier, by setting it to the smallest natural class that contains the segments from both X and Y. For example for the data in (20a)—surface forms reproduced in (35)—if the absence of non-harmonizing sibilant trigrams is sufficiently statistically conspicuous, then the Hayes and Wilson (2008) model may learn the constraints *[s][][ʃ] and *[ʃ][][s].

(35) [ʃokuʃiʃ]
[apʃaʃ]
[ʃuniʃ]
[sokis]
[sigosis]
[uts]

The smallest natural class containing both [s] and [ʃ] is [+stri], so Gouskova and Gallagher (2020)'s model would then project the [+stri] tier and re-apply Hayes and Wilson (2008) over that projection. The success of this model depends upon the trigram restriction being clear enough in the data to discover the relevant projection, and upon the effectiveness of Hayes and Wilson (2008) at discovering constraints over the projection. Moreover, the authors recognized a limitation: the model's inability to capture more complex phenomena like opaque or blocking segments, which must be included on the tier but do not participate in the restriction. This is because the model constructs the tier based on the sets X and Y, which do not contain information about blocking segments.

Goldsmith and Riggle (2012) proposed an information theoretic model for justifying a tier-based descriptive account of Finnish vowel harmony. They used Goldsmith and Xanthos (2009)'s Hidden Markov Model (HMM) approach to extract the two classes of segments that maximize the probability of the data. Because the segments of a word tend to alternate between consonants and vowels (e.g. CVCV is much more frequent than CCVV), this HMM approach is best-suited for extracting the two categories *consonant* and *vowel*. Goldsmith and Riggle (2012) used a Boltzmann model to score phonological ill-formedness in terms of unigram probabilities, and bigram mutual information over the surface string and the vowel tier. The intuition for the model is that it combines the frequency of segments (unigrams) with the frequency of bigrams over the words and vowel-tier projections to score phonological ill-formedness. This model is largely limited to interactions between all vowels or all consonants, because the HMM usually constructs the consonant and vowel categories. Thus, on (35), the model could find dependencies between sibilants on the consonant tier, but non-sibilant consonants would prevent some sibilants from occurring withing the purview of the bigram mutual-information computation.

3.2 Formal-Language-Theoretic Approaches

Heinz (2010) proposed a model for learning long-distance phonotactics based on the *precedence* relation. However some long-distance patterns cannot be accounted for in terms of the precedence relation (Heinz, 2010; Jardine and Heinz, 2016). Consequently, later formal-language-theoretic approaches have instead targeted the class of Tier Strictly-Local (TSL) constraints Heinz *et al.* (2011), which have been argued to subsume most or all non-local consonant interactions (McMullin, 2016). Their functional analogue, the TSL functions, are argued to cover a broad range of non-local processes (Burness *et al.*, 2021).

In particular, Jardine and Heinz (2016) proposed a model for learning Tier-based Strictly 2-Local (TSL_2) formal languages, which are languages where the words of the language can be distinguished from the non-grammatical by a tier-sequence of length 2. The model is provably capable of learning such languages, in the sense of Gold (1967), and, like D2L, iteratively removes segments from a tier that initially contains all segments. Jardine and McMullin (2017) extended the model to handle arbitrary values of k, and Lambert (2021) demonstrated that TSL languages are also learnable in an online setting. Jardine (2016b) applied Jardine and Heinz (2016)'s model to idealized natural language data, showing that the model successfully learns phonotactic restrictions in the setting where exceptions were removed and segments were pre-organized into natural classes.

Of particular importance to the present paper is the model proposed by Burness and McMullin (2019) for learning Output Tier-Strictly 2-Local (OTSL_2) functions. While most prior work has dealt with phonotactic learning, Burness and McMullin's focus, along with ours, is on alternations. Moreover, Burness and McMullin's model shares some core characteristics with D2L. Burness and McMullin start with the definition of OTSL_2 functions and prove properties of that class of functions. These properties, together with the fact that output-strictly-local functions are a special case with all segments on the tier (§ 2.4), lead to the development of a learning algorithm that starts with all segments on the tier and iteratively removes those that cannot be on the tier. Burness and McMullin (2021) improved the algorithm's runtime.

By starting with a formal characterization of phonological phenomena and working out what properties an algorithm for learning such structures must have, formal-language-theoretic approaches attempt to solve a part of the learning problem from an analytical perspective, and are not necessarily intended for learning directly on naturalistic language data. We view the fact that D2L is derived from independent psychological mechanisms, is successful on naturalistic language data, and uses an analogous iterative change of representation, as a desirable result for for the overall approach.

3.3 Neural Network Models

There have been several attempts to model aspects of vowel harmony with recurrent neural networks (RNNs). Hare (1990) proposed using an RNN to model Hungarian vowel harmony, training it on synthetic bit-sequences and finding that its assimilatory behavior on these bit-sequences mirrored some of the complexities of Hungarian vowel harmony. Rodd (1997), while not targeting vowel harmony directly, found that RNN models could use distributional information to learn phonological categories from a small Turkish corpus, and that they could learn to treat [+back] and [−back] vowels differently.

These early models were limited in their empirical scope, but in the decades since, research on neural networks (NNs) has accelerated. NNs have been used for morphological reinflection (Cotterell *et al.*, 2016) and to revisit connectionism in the 'past-tense debate' of English morphology (Kirov and Cotterell, 2018). While NNs as cognitive models of morphophonological learning has been questioned (e.g., McCurdy *et al.* 2020; Belth *et al.* 2021) and these more recent models have not directly been applied to modeling long-distance alternations, some of the languages included in the SIGMORPHON reinflection task involve non-local dependencies. Moreover, Smith *et al.* (2021) used an RNN-based model to evaluate an articulatory account of height harmony in Nzebi by training the model on simulated speech to map segments to vocal-tract articulator movements.

Because prior works' problem settings have varied, it is difficult to draw conclusions about RNNs' ability to model non-local alternations, but an RNN is certainly applicable as a comparison model for D2L (see § 4.1.1 and § A.3 for details of how we use it as such).

4 Comparison to Human Behavior

4.1 Model Behavior on Finley (2011)

We first compare to Finley (2011)'s artificial language experiment. Finley's experiment presented participants with training data consisting of <stem, suffixed> pairs. The suffix contained a sibilant that harmonized with a stem sibilant across an intervening vowel (36).

(36) /dis̱o-s̱u/ → [dis̱os̱u]
/nes̱i-s̱u/ → [nes̱is̱u]
/piʃ̱a-s̱u/ → [piʃ̱aʃ̱u]
/kuʃ̱o-s̱u/ → [kuʃ̱oʃ̱u]

To make the sibilants adjacent on a tier, the vowels must be excluded, suggesting [+cons] is the relevant tier. This predicts that when vowels and non-sibilant consonants intervene, the non-sibilant consonants will block the harmony. This prediction is borne out. After training, participants were evaluated in a two-alternative forced choice (2AFC) paradigm, where they were presented with a stem and two choices for its suffixed from: one harmonizing and one not. Learners generalized to novel instances like the training instances (37a), demonstrating that they learned a harmony pattern. However, they showed no preference for harmony in novel cases where non-sibilant consonants also intervened (37b), choosing the non-harmonizing option as often as a control group did.

(37) a. /bas̱o-s̱u/ → ✓ [bas̱os̱u], *[bas̱oʃ̱u]
/deʃ̱e-s̱u/ → ✓ [deʃ̱eʃ̱u], *[deʃ̱es̱u]

b. /ʃ̱eta-s̱u/ → ? [ʃ̱etaʃ̱u], ? [ʃ̱etas̱u]
/ʃ̱omi-s̱u/ → ? [ʃ̱omiʃ̱u], ? [ʃ̱omis̱u]

In contrast, a second experiment presented participants with training data where sibilants harmonize across both intervening vowels and non-sibilant consonants (38).

(38) /s̱uge-s̱u/ → [s̱uges̱u]
/s̱one-s̱u/ → [s̱ones̱u]
/ʃ̱upe-s̱u/ → [ʃ̱upeʃ̱u]
/ʃ̱ako-s̱u/ → [ʃ̱akoʃ̱u]

In this case, the [+sib] tier is needed in order for the sibilants to be adjacent, in which case learners should generalize to cases where only vowels intervene. The prediction is again borne out. Learners generalized to novel train-like instances (39a), and instance where only a vowel intervened (39b).

(39) a. /ʃ̱ika-s̱u/ → ✓ [ʃ̱ikaʃ̱u], *[ʃ̱ikas̱u]
/ʃ̱ege-s̱u/ → ✓ [ʃ̱egeʃ̱u], *[ʃ̱eges̱u]
b. /keʃ̱u-s̱u/ → ✓ [keʃ̱uʃ̱u], *[keʃ̱us̱u]
/niʃ̱a-s̱u/ → ✓ [niʃ̱aʃ̱u], *[niʃ̱as̱u]

4.1.1 Comparison Models

GR is Goldsmith and Riggle (2012)'s phonotactic model (see also § 3 and § 6). To compare the relative well-formadness of candidates, we use the Boltzmann score from pg. 882 of their paper. Following the authors, we used Laplace smoothing with 0.5 smoothing-factor.

GG is Gouskova and Gallagher (2020)'s phonotactic model (see § 3 and § 6 for further description). We use the author's code[8] and default parameters.

TSLIA is Jardine and Heinz (2016); Jardine and McMullin (2017)'s formal-language-theoretic model. We used the implementation from Aksënova (2020). This model is binary: it accepts or rejects a candidate string. When more than one candidate is accepted by the model, we choose one at random as the best candidate. We used TSLIA instead of Burness and McMullin (2019)'s model because we are not aware of an implementation of the latter.

LSTM is a Recurrent Neural Network (RNN) sequence-to-sequence model. We trained the model to predict the surface form of each underlyingly underspecified segment. This

[8] https://github.com/gouskova/inductive_projection_learner

simplifies the learning problem by not requiring the model to predict the surface form for the entire sequence, but makes for a fair comparison to D2L, which also has access to the underlying forms. We used a Pytorch (Paszke *et al.*, 2019) implementation, and discuss architecture specifics, training procedure, and hyperparameter tuning in § A.3.

3G is a trigram phonotactic model, which assigns a probability to each candidate in terms of the trigrams it contains—how frequent they are in the training data. As in GR, we used Laplace smoothing with a smoothing-factor of 0.5.

4.1.2 Setup

We use the training and test items recorded in Finley (2011, p. 15)'s appendix. Each model is trained on the training instances, then probed to choose between each pair of items in the 2AFC test set. We treat the sibilant in [-su]/[-ʃu] as underlyingly /S/, unspecified for [ant]. The comparison phonotactic models are trained on the surface forms, and, for each test trial, the model assigns a score to the [-su] and [-ʃu] forms and the form with the higher score is chosen. If the model assigns the same score to both items, then the choice is made at random. Since D2L produces an output for an input, we use its produced form as its choice if the produced form matches one of the 2AFC choices. Otherwise the choice is made at random. To simulate multiple participants, we run each model 30 times and report averages and standard deviations. This is important because some randomness is introduced due to GR, GG, and 3G being stochastic, and because when a model assigns the same score to both 2AFC choices, the choice is made at random. Following Finley (2011), each of the 24 training items appears 5 times in the total exposure set, and the items are presented in a random order for each of the 30 runs. We list the segment features in Tab. 8 in the appendix.

4.1.3 Results

The results for the first experiment are given in Tab. 1, and for the second experiment in Tab. 2. The tables report, for each model, the fraction of the test instances where the model picked the harmonizing choice. The Hum row records a '✓' whenever Finley (2011) reported that the experimental-group participants chose the harmonizing choice significantly more often than the control-group participants. It records an '✗' wherever the two groups chose the harmonizing choice at statistically indistinguishable rates.[9] If, over a set of test

[9] We chose to report the human results in this way because Finley (2011) only reported these conclusions, not the actual mean rates of the harmonizing choice.

Table 1: Results for Finley (2011)'s first experiment, where training instances involved sibilants harmonizing across intervening vowels. Test instances are of three types: train (Old), novel train-like (New Train-Like), and novel items where both vowels and non-sibilant consonants intervene between sibilants (Novel). D2L generalizes in exactly the cases where humans do, and does not generalize in exactly the cases where humans do not.

	Train CV$\underline{\text{S}}$V-$\underline{\text{S}}$V		
	CV$\underline{\text{S}}$V-$\underline{\text{S}}$V (Old)	CV$\underline{\text{S}}$V-$\underline{\text{S}}$V (New Train-Like)	$\underline{\text{S}}$VCV-$\underline{\text{S}}$V (Novel)
Hum	✓	✓	✗
D2L	1.0000 ± 0.00	1.0000 ± 0.00	0.3333 ± 0.00
GG	0.5389 ± 0.13	0.5394 ± 0.16	0.4500 ± 0.14
GR	0.4667 ± 0.27	0.6364 ± 0.18	0.5333 ± 0.07
LSTM	0.8917 ± 0.17	0.7970 ± 0.18	0.8611 ± 0.25
TSLIA	0.5389 ± 0.13	0.5394 ± 0.16	0.4500 ± 0.14
3G	1.0000 ± 0.00	1.0000 ± 0.00	0.7056 ± 0.04

items, a model makes the harmonizing choice significantly more often than a control model that makes a random selection from the two choices, then we treat the model as having generalized to those test items. The test of significance is made with a one-sided t-test that compares the average model performance over 30 runs to that of the random control model. The null hypothesis is that the tested model's average performance is equal to the random control model's. We use Welch's t-test, which does not assume equal variance, and a significance level of $\alpha = 0.99$. We shade a cell gray if the model matches the human result—i.e. iff either both Hum and the model generalized or neither generalized.

4.1.4 Discussion

D2L matches the human results in all cases. When trained on CV$\underline{\text{S}}$V-$\underline{\text{S}}$V words, D2L learns a [+cons] tier (40a), and thus produces harmony in novel CV$\underline{\text{S}}$V-$\underline{\text{S}}$V (second column, Tab. 1), but not $\underline{\text{S}}$VCV-$\underline{\text{S}}$V words (third column, Tab. 1). In contrast, when trained on $\underline{\text{S}}$VCV-$\underline{\text{S}}$V words, D2L learns a [+stri] tier (40b), which produces harmony in both novel $\underline{\text{S}}$VCV-$\underline{\text{S}}$V words (second column, Tab. 2) and novel CV$\underline{\text{S}}$V-$\underline{\text{S}}$V words (third column, Tab. 2).

(40) a. Agree({S}, {ant})/[+cons] __ ∘ proj(·, [+cons])
b. Agree({S}, {ant})/[+stri] __ ∘ proj(·, [+stri])

D2L's performance is categorical because, as an algorithm, it abstracts away from experimental complexities, like the fact that human participants may lose attention or not un-

Table 2: Results for Finley (2011)'s second experiment, where training instances involved sibilants harmonizing across both vowels and non-sibilant consonants. Test items are of three types: train (Old), novel train-like (New Train-Like) and novel items where only vowels intervene between sibilants (Novel). Both humans and D2L learned a harmony pattern (Old) and extend it to both New Train-Like and Novel test instances.

	Train S̲VCV-S̲V		
	S̲VCV-S̲V (Old)	S̲VCV-S̲V (New Train-Like)	CVS̲V-S̲V (Novel)
Hum	✓	✓	✓
D2L	1.0000 ± 0.00	1.0000 ± 0.00	1.0000 ± 0.00
GG	0.4944 ± 0.16	0.4778 ± 0.16	0.4667 ± 0.14
GR	1.0000 ± 0.00	0.4167 ± 0.00	0.2500 ± 0.00
LSTM	0.9167 ± 0.19	0.8889 ± 0.30	0.7139 ± 0.32
TSLIA	0.5444 ± 0.15	0.4944 ± 0.16	0.4500 ± 0.14
3G	1.0000 ± 0.00	0.5833 ± 0.00	0.2500 ± 0.00

derstand the task.

The comparison models are largely ineffective at generalizing at all from the limited training data, with the exception of LSTM. However, when trained on CVS̲V-S̲V words, LSTM generalizes to both CVS̲V-S̲V and S̲VCV-S̲V test words, whereas humans do not generalize to S̲VCV-S̲V words (Tab. 1). Thus, LSTM fails to exhibit the blocking behavior of non-sibilant consonants in the first experiment.

GR is able to achieve moderate performance generalizing to CVS̲V-S̲V words when trained on words of the same type (second column, Tab. 1), and for S̲VCV-S̲V words, the intervening C prevents the sibilants from being adjacent on the consonant tier that the Hidden Markov Model learns (third column, Tab. 1). However, when trained on S̲VCV-S̲V, no interactions between sibilants are visible to the model and it is unable to generalize beyond the training data (Tab. 2).

GG appears unable to generalize from these small datasets. While this may seem to be a limitation of the data, note that humans were able to generalize from the same training data. In the next section, we turn to McMullin and Hansson (2019)'s study, which involves substantially more data, and is thus more instructive regarding GG's generalization behavior.

TSLIA's performance indicates that the data does not contain the model's characteristic sample. 3G frequently chooses the harmonizing form for S̲VCV-S̲V words when trained on CVS̲V-S̲V words, and rarely chooses the harmonizing form for CVS̲V-S̲V words when trained on S̲VCV-S̲V words. This is the opposite of what humans did.

4.2 Model Behavior on McMullin and Hansson (2019)

McMullin and Hansson (2019) replicated results similar to Finley (2011)'s, extending them to both liquid harmony and dissimilation. In these experiments the harmony/dissimilation was *regressive*. Participants were first exposed to a practice phase where they were presented with verb stems followed by a past-tense form, which added a [-ɹu] suffix to the stem, and the same stems followed by a future-tense form, which added a [-li] suffix to the stem. These practice-phase stems did not contain liquids: they allowed the participants to learn the relevant morphology.

Next, in the training phase, participants were presented with <stem, past, future> triplets. The two training settings of Finley (2011) where doubled by McMullin and Hansson (2019), performing each experiment in both assimilatory and dissimilatory settings. In all experiments, the data contained 50% distractors, which were stems with no liquid.

In what the authors labeled experiment 1a, the stem liquid harmonized regressively with the affix liquid across an intervening vowel (41) (treating the alternating stem-liquid as underlyingly underspecified /L/).

(41) /toboL̲e-ɹ̲u/ → [toboɹ̲eɹ̲u]
/toboL̲e-l̲i/ → [tobol̲el̲i]
/dumiL̲i-ɹ̲u/ → [dumiɹ̲iɹ̲u]
/dumiL̲i-l̲i/ → [dumil̲il̲i]

Symmetrically, in experiment 2a, the stem liquids *dissimilated* regressively from the affix liquid across an intervening vowel (42).

(42) /toboL̲e-ɹ̲u/ → [tobol̲eɹ̲u]
/toboL̲e-l̲i/ → [toboɹ̲el̲i]
/dumiL̲i-ɹ̲u/ → [dumil̲iɹ̲u]
/dumiL̲i-l̲i/ → [dumiɹ̲il̲i]

Like Finley (2011)'s study, participants generalized the training pattern to novel words of the same CVCVL̲V-L̲V form, but did not extend it to CVL̲VCV-L̲V or L̲VCVCV-L̲V forms, where the liquids crossed more than just vowels. This is the predicted behavior if participants construct a [+cons] tier.

In experiments 1b and 2b, the participants were presented with assimilatory (43a) and dissimilatory (43b) instances of the form CVL̲VCV-L̲V.

Table 3: Results from McMullin and Hansson (2019)'s Experiment 1a (assimilation) and 2a (dissimilation), where training instances involved liquids interacting across intervening vowels. D2L matches human behavior in all cases.

	Train CVCVḺV-ḺV (assimilation)			Train CVCVḺV-ḺV (dissimilation)		
	CVCVḺV-ḺV	CVḺVCV-ḺV	ḺVCVCV-ḺV	CVCVḺV-ḺV	CVḺVCV-ḺV	ḺVCVCV-ḺV
Hum	✓	✗	✗	✓	✗	✗
D2L	1.000 ± 0.00	0.394 ± 0.07	0.471 ± 0.05	1.000 ± 0.00	0.419 ± 0.07	0.467 ± 0.05
GG	0.477 ± 0.10	0.617 ± 0.08	0.493 ± 0.06	1.000 ± 0.00	0.546 ± 0.04	0.475 ± 0.02
GR	0.972 ± 0.08	0.652 ± 0.07	0.495 ± 0.05	0.996 ± 0.02	0.368 ± 0.06	0.510 ± 0.05
LSTM	0.850 ± 0.23	0.840 ± 0.22	0.795 ± 0.22	0.833 ± 0.24	0.831 ± 0.23	0.820 ± 0.23
TSLIA	0.503 ± 0.09	0.520 ± 0.09	0.489 ± 0.07	0.497 ± 0.09	0.480 ± 0.09	0.511 ± 0.07
3G	0.969 ± 0.00	0.637 ± 0.06	0.505 ± 0.09	1.000 ± 0.00	0.362 ± 0.06	0.495 ± 0.09

(43) a. /teḺomu-ɹ̱u/ → [teɹ̱omuɹ̱u]
/teḺomu-ḻi/ → [teḻomuḻi]
/poḺeku-ɹ̱u/ → [poɹ̱ekuɹ̱u]
/poḺeku-ḻi/ → [poḻekuḻi]

b. /teḺomu-ɹ̱u/ → [teḻomuɹ̱u]
/teḺomu-ḻi/ → [teɹ̱omuḻi]
/poḺeku-ɹ̱u/ → [poḻekuɹ̱u]
/poḺeku-ḻi/ → [poɹ̱ekuḻi]

A liquid tier is needed for the training liquids to be adjacent, predicting that learners will extend the pattern to CVCVḺV-ḺV and ḺVCVCV-ḺV forms. These predictions were borne out.

4.2.1 Setup

We follow McMullin and Hansson (2019, sec. 2)'s setup to produce the stimuli. The setup details are the same as in the prior experiment (§ 4.1.2) except that, following McMullin and Hansson (2019), the stimuli are only presented once each. The stimuli include the practice phase forms. We treat alternating stem liquids [l]/[ɹ] as underlyingly underspecified /L/. We use the same comparison models as in the prior experiment § 4.1.1 and list the segment features in Tab. 9 in the appendix.

Table 4: Results from McMullin and Hansson (2019)'s. Experiment 1b (assimilation) and 2b (dissimilation), where training instances involved liquids interacting across intervening vowels. D2L matches human behavior in all cases.

	Train CVL̲VCV-L̲V (assimilation)			Train CVL̲VCV-L̲V (dissimilation)		
	CVCVL̲V-L̲V	CVL̲VCV-L̲V	L̲VCVCV-L̲V	CVCVL̲V-L̲V	CVL̲VCV-L̲V	L̲VCVCV-L̲V
Hum	✓	✓	✓	✓	✓	✓
D2L	1.000 ± 0.00	1.000 ± 0.00	1.000 ± 0.00	1.000 ± 0.00	1.000 ± 0.00	1.000 ± 0.00
GG	0.512 ± 0.04	0.497 ± 0.04	0.495 ± 0.04	1.000 ± 0.00	1.000 ± 0.00	1.000 ± 0.00
GR	0.455 ± 0.09	0.641 ± 0.08	0.503 ± 0.06	0.560 ± 0.06	0.359 ± 0.08	0.499 ± 0.05
LSTM	0.767 ± 0.25	0.767 ± 0.25	0.767 ± 0.25	0.800 ± 0.24	0.800 ± 0.24	0.800 ± 0.24
TSLIA	0.503 ± 0.09	0.520 ± 0.09	0.489 ± 0.07	0.497 ± 0.09	0.480 ± 0.09	0.511 ± 0.07
3G	0.440 ± 0.06	0.641 ± 0.08	0.502 ± 0.10	0.560 ± 0.06	0.359 ± 0.08	0.498 ± 0.10

4.2.2 Results

The results for the experiments with CVCVL̲V-L̲V exposure data are given in Tab. 3, and the results for the experiments with CVL̲VCV-L̲V exposure data in Tab. 4. For the assimilation experiments, the tables report, for each model, the fraction of the test instances where the model picked the harmonizing choice. For the dissimilatory experiments, they report the fraction where the disharmonizing choice was made. The Hum row records a '✓' whenever McMullin and Hansson (2019) reported that the experimental-group participants extended the training pattern to test instances of the form in the corresponding column, and an '✗' where they did not. Gray cells mark where the models match the human result. As before, the models are compared to a control model that makes a random selection in the 2AFC test.

4.2.3 Discussion

D2L matches the human results in every setting, and is the only model to do so. When trained on CVCVL̲V-L̲V words, D2L learns (44a) and (44c), with a [+cons] tier projection. When trained on CVL̲VCV-L̲V words, D2L learns (44b) and (44d), with a {l, ɹ, L} liquid tier. These results also demonstrate that D2L is able to construct regressive rules and infer whether a process is assimilatory or dissimilatory (see § 2.3.4).

(44) a. $\textsc{Agree}(\{\text{L}\}, \{\text{ant}, \text{cor}, \text{lat}\}) /\ \_\_\ [+\text{cons}] \circ \text{proj}(\cdot, [+\text{cons}])$

b. $\textsc{Agree}(\{\text{L}\}, \{\text{ant}, \text{cor}, \text{lat}\}) /\ \_\_\ \{\text{l}, \text{ɹ}\} \circ \text{proj}(\cdot, \{\text{l}, \text{ɹ}, \text{L}\})$

c. $\textsc{Disagree}(\{\text{L}\}, \{\text{ant}, \text{cor}, \text{lat}\}) /\ \_\_\ [+\text{cons}] \circ \text{proj}(\cdot, [+\text{cons}])$

d. $\textsc{Disagree}(\{\text{L}\}, \{\text{ant}, \text{cor}, \text{lat}\}) /\ \_\_\ \{\text{l}, \text{ɹ}\} \circ \text{proj}(\cdot, \{\text{l}, \text{ɹ}, \text{L}\})$

When trained on CVCV$\underline{\text{L}}$V-$\underline{\text{L}}$V words (Tab. 3), D2L is the only model to match human behavior in all cases, though some comparison models also come close. However, when trained on CV$\underline{\text{L}}$VCV-$\underline{\text{L}}$V words (Tab. 4), only D2L and LSTM come close to human behavior. Similarly to the previous experiment (§ 4.1), when trained on CVCV$\underline{\text{L}}$V-$\underline{\text{L}}$V words, LSTM generalizes to words where non-liquids intervened and humans did not generalize (Tab. 3). These results suggest that LSTM may not be able to express blockers, instead learning a dependence between L's independent of what segments intervene.

Because the HMM of GR usually learns the consonant/vowel tiers, the interacting liquids in CV$\underline{\text{L}}$VCV-$\underline{\text{L}}$V words are blocked by the intervening C, which prevents the model from substantially generalizing. Similarly, the interacting liquids are beyond the trigram sensitivity of 3G, so its occasional above-chance performance is only due to coincidental statistical regularities in the exposure data unrelated to the liquid interaction. TSLIA's performance again indicates that the data does not contain the model's characteristic sample.

The asymmetry of GG between assimilation and dissimilation deserves some discussion. Because no -LVL- sequences occur in CV$\underline{\text{L}}$VCV-$\underline{\text{L}}$V training instances, the Hayes and Wilson (2008) model extracts a trigram constraint *[l,ɹ][][l,ɹ], which GG uses to project a liquid tier. In the assimilation experiment, GG learns the single constraint *[l,ɹ][l,ɹ] on this tier, not two constraints *[lɹ] and *[ɹl]. Consequently, it cannot distinguish between assimilitory and non-assimilatory sequences. In the dissimilation experiment, GG learns the single tier-constraint *[ll], as well as a non-tier constraint *[−cor], which applies to [ɹ] but not [l]. Consequently [ɹl] » [ll] due to the tier constraint, which has a much higher weight than the non-tier constraint. Moreover, [lɹ] » [ɹɹ] because the latter violates *[−cor] twice. Thus, the two constraints coincidentally conspire to yield a functional generalization. However, the coincidental result only works for dissimilation, and the asymmetry with assimilation is not consistent with human behavior, which was similar for assimilation and dissimilation.

5 Learning Natural Language Alternations

5.1 Turkish Vowel Harmony

The Turkish vowel inventory, (45), has 4 [+back] and 4 [−back] vowels, all of which participate in [±back] harmony (Kabak, 2011, p. 2).

(45)

	front		back	
	unround	round	unround	round
high	i	y	ɯ	u
low	e	ø	ɑ	o

Affix vowels alternate between [+back] vowels when the final stem vowel is [+back] and [−back] when the final stem vowel is [−back], as shown in (46) repeated from (2).

(46)	[dɑ̲l-lɑ̲r-ɯ̲n]	branch-PL-GEN	(Kabak, 2011, p. 3)
	[je̲r-le̲r-i̲n]	place-PL-GEN	(Kabak, 2011, p. 3)
	[i̲p-le̲r-i̲n]	rope-PL-GEN	(Nevins, 2010, p. 28)

In addition to back/front harmony, the [+high] vowels participate in secondary rounding harmony, as exemplified by the GEN affix (47). The [+high] vowels harmonize with the final vowel of the stem, both when the the stem vowel is [+high] (47a) and [−high] (47b).

(47)	a.	[i̲p-i̲n]	rope-GEN	(Nevins, 2010, p. 29)
		[jy̲z-y̲n]	face-GEN	(Kabak, 2011, p. 3)
		[kɯ̲z-ɯ̲n]	girl-GEN	(Nevins, 2010, p. 29)
		[bu̲z-u̲n]	ice-GEN	(Kabak, 2011, p. 3)
	b.	[e̲l-i̲n]	hand-GEN	(Kabak, 2011, p. 3)
		[sø̲z-y̲n]	word-GEN	(Nevins, 2010, p. 29)
		[sɑ̲p-ɯ̲n]	stalk-GEN	(Nevins, 2010, p. 29)
		[jo̲l-u̲n]	road-GEN	(Kabak, 2011, p. 3)

We follow Nevins (2010) and Belth (2023c) in treating primary and secondary harmony as a single process, where alternating low vowels are underspecified for their [back] value underlyingly, and alternating [+high] vowels are underspecified for both [back] and [round]. Consequently, Agree is taken to copy only the underspecified vocalic features from the closest vowel, i.e., both [back] and [round] for [?back, +high, ?round], but only [back] for [?back, ±high, ±round]. The fact that alternating [+high] vowels take their [round] value from *any* vowel, regardless of height, is evidence in favor of this treatment. If we were to treat primary and secondary harmony as separate processes, D2L could be run twice—once for feature [±back] and once for [±round]—to construct two generalizations.

Some affixes do not participate in vowel harmony, and some are “half-harmonizing”, meaning that one of two vowels participates. These exceptions do not alternate on the surface, so there is no motivation for underspecification (Nevins, 2010, sec. 2.6).

Some Turkish suffixes, such as the locative, also exhibit a local voicing assimilation process, where an affix-initial alveolar stop matches the voicing of the preceding segment (48; examples from Dobrovolsky 1982; Çöltekin 2010; Kornfilt 2013).

(48)	[byro-dɑ]	‘office’-LOC
	[ev-de]	‘house’-LOC
	[ʤep-te]	‘pocket’-LOC

5.1.1 Setup

We use two datasets, created by Belth (2023c) from CHILDES and MorphoChallenge (Kurimo *et al.*, 2010) respectively. Both datasets contain frequency-annotated, IPA-transcribed surface forms. The CHILDES dataset contains 1,727 word types and the MorphoChallenge dataset contains 22,315. We used morphological analyses available in the datasets to construct underlying forms. We set the underlying form of each segment in each morpheme to match its surface form unless the segment appears in different forms across realizations of the morpheme. For alternating segments, we set any feature that alternated on the surface as underlyingly unspecified. For instance the vowel of the PL affix [-lɑr]/[-ler] is set to /?back, −high, −round/, and the GEN affix vowel [-in]/[-yn]/[-ɯn]/[-un] is set to /?back, +high, ?round/. This process results in underspecified URs /A/ with extension {ɑ, e}, /H/ with extension {i, y, ɯ, u}, and /D/ with extension {d, t}. We list the segment features in Tab. 10 in the appendix.

For training data, we sampled 1K unique words, weighted by frequency, and used the remaining of words for testing (727 for CHILDES and 21,315 for MorphoChallenge). We repeated this 30 times with a different random sample each time. We report the average performance across the 30 runs. The comparison phonotactic models are trained on the surface forms, and each test surface form is predicted by computing all possible specifications of the input UR’s underspecified segments and choosing the form that the model assigns the highest score to. For instance, for input /dɑl-lAr-Hn/, the candidate surface forms are [dɑl-lɑr-in], [dɑl-lɑr-yn], [dɑl-lɑr-ɯn], [dɑl-lɑr-un], [dɑl-ler-in], [dɑl-ler-yn], [dɑl-ler-ɯn], [dɑl-ler-un]. The comparison models are the same as the prior experiments.

Since the data involves both vowel harmony and local voicing assimilation, D2L seeks

Table 5: Accuracy of models on held-out test words, when learning Turkish vowel harmony.

	Test Accuracy	
Model	CHILDES	MorphoChallenge
D2L	**0.9955 ± 0.00**	**0.9872 ± 0.00**
GG	0.9628 ± 0.02	0.9083 ± 0.05
GR	0.9810 ± 0.01	0.9291 ± 0.08
LSTM	0.8866 ± 0.07	0.8290 ± 0.13
TSLIA	0.6421 ± 0.01	0.6388 ± 0.00
3G	0.8213 ± 0.01	0.7421 ± 0.01

to create a generalization for each process. This is accomplished automatically (i.e., not hard-coded) by following Belth (2023a) in running the search for a generalization whenever a new discrepancy is found (e.g., /A/ → [e] or /D/ → [t]).

5.1.2 Results

The results shown in Tab. 5 demonstrate that D2L learns generalizations that robustly predict the surface form of unseen test words. D2L's accuracy is higher than the comparison models and is consistent with acquisition studies, which reveal that Turkish-speaking children as young as 2;0—when their vocabulary likely contains no more than a thousand words—already know vowel harmony well enough to extend it to nonce words (Altan, 2009). D2L's errors are due to genuine exceptions, which it is able to tolerate and still find a generalization by lexicalizing these under the Tolerance Principle. For example, D2L predicts *[saatlar] for the word [saat-ler] 'watch-PL', which originates from Arabic and exceptionally does not harmonize in adult speech. Like D2L, children appear to overextend vowel harmony to exceptional words (Altan, 2009). The rules D2L constructed are (49).

(49) a. $\textsc{Agree}([?\text{back}], \{\text{back}, \text{round}\}) / [-\text{cons}]\ \_\_ \circ \text{proj}(\cdot, [-\text{cons}])$

b. $\textsc{Agree}([?\text{voice}], \{\text{voice}\}) / [*]\ \_\_$

The vowel harmony rule (49a) states that vowels with unspecified backness (i.e., [?back]) take their [back] value and, if also unspecified [?round], their [round] value from a vowel to the left on a projected vowel tier. The voicing assimilation rule (49b) matches /D/'s voicing to any segment (denoted by [*]) to its left. We omit the projection component from this rule since it trivially maps all segments, which demonstrates the D2L does not change representations when string-adjacent dependencies can predict the alternation (§ 2.4). The generalizations are consistent with standard descriptive analyses (Kabak, 2011).

5.2 Finnish Vowel Harmony

The Finnish vowel inventory (50) (Suomi *et al.* 2008, sec. 3.1) has 8 vowels, 6 of which participate in front/back harmony, as discussed in § 1 and repeated in (51).

		front		back	
		unround	round	unround	round
(50)	high	i	y		u
	mid	e	ø		o
	low	æ		ɑ	

The vowels {i, e} are neutral, neither participating in nor blocking harmony (51b). When a stem contains only neutral vowels, alternating affix vowels are [−back] by default (51c).

(51)	a.	[pøytæ̲-næ̲]	table-ESS	(Ringen and Heinämäki, 1999, p. 304)
		[poutɑ̲-nɑ̲]	fine weather-ESS	(Ringen and Heinämäki, 1999, p. 304)
	b.	[ko̲ti-nɑ̲]	home-ESS	(Ringen and Heinämäki, 1999, p. 305)
	c.	[velje-næ]	road-ESS	(Nevins, 2010, p. 76)

Sinc the neutral vowels are [−back], the default case (51c) could instead be characterized as the affix vowel harmonizing with the neutral vowels only when no non-neutral vowels are present. However, the analysis with a default value is preferred, because it extends to Uyghur, which has the same vowel organization, but alternating vowels are [+back] when the stem contains only [−back] neutral vowels (Lindblad 1990; Nevins 2010, p. 77-78).

5.2.1 Setup

Finnish data comes from the MorphoChallenge (Kurimo *et al.*, 2010), which includes 1835 frequency-annotated words with morphological segmentations. The surface form of these affixes varied extensively in ways beyond just vowel alternations (e.g., the GEN affix has forms [n], [en], [ten], ...) making it challenging to compute precisely how vowels alternate. Consequently, we treated all affix vowels as alternating and underlyingly unspecified for feature [back]. This choice potentially overestimates the number of harmony exceptions, and thus potentially makes the problem more challenging for D2L, while not affecting the comparison phonotactic models, which learn directly from surface forms. We dropped words with only one occurrence, yielding a set of 1219 words. Finnish orthographic vowels map directly to phonemes. We mapped orthographic consonants to phonemes directly, but treated <x> as [ks] and <c>, <q> as [k]. The segment features are in Tab. 12 in the appendix.

For training data, we sampled 80% of words (975) weighted by frequency and used the other 20% for testing. We repeated this sampling 30 times, and report the average performance across the 30 runs. To to get a predicted surface form for the phonotoactic comparison models, we again computed candidate surface forms by permuting affix vowels between back and front, and selecting the candidate that the model assigns the highest score to.

5.2.2 Results

The accuracies shown in Tab. 6 reveal that D2L is the most accurate model. D2L learns the generalization (52), where unspecified vowels take their [±back] feature from the vowel to the left, skipping consonants and neutral vowels (52a). D2L also learned that if no non-neutral vowel is to the left, the surface vowel should be [−back] (52b).

(52) a. $\textsc{Agree}([?\text{back}], \{\text{back}\}) / [-\text{cons}] \_\_ \circ \text{proj}(\cdot, [-\text{cons}] \setminus \{\text{i}, \text{e}\})$
b. Elsewhere [−back]

The tier correctly excludes the neutral vowels while preserving all other vowels. On a small number of simulations, the tier contained one or two spurious segments—[f] and/or [b]—because they never occur between an affix vowel and a stem vowel in the training data and, consequently, never interfere with harmony. The elsewhere condition matches the default value given in accounts of Finnish for what happens when a stem contains only neutral vowels (Ringen and Heinämäki, 1999). Note that because the segments do not fall neatly into a natural class, D2L simply listed the tier segments explicitly; we presented it here with natural classes and set operations for clarity.

5.3 Latin Liquid Dissimilation

In Latin, default /l/ (5a) dissimilates to [r] when preceded by /l/ across varying distances, as shown in (5b) and repeated in (53b). This process is usually discussed and most clearly seen in the adjectival *-alis*/*-aris* affix. The dissimilation of /l/ is blocked by intervening /r/ (53c). Cser (2010) argues that intervening [−cor] consonants also block dissimilation (53d).

Table 6: Accuracy of models on held-out test words, when learning Finnish vowel harmony.

Model	Test Accuracy
D2L	**0.9799 ± 0.01**
GG	0.8113 ± 0.03
GR	0.8441 ± 0.03
LSTM	0.8070 ± 0.06
TSLIA	0.8350 ± 0.02
3G	0.8418 ± 0.02

(53)	a.	*nav-alis*	‘naval’
	b.	*popul-aris*	‘popular’
		lun-aris	‘lunar’
	c.	*flor-alis*	‘floral’
	d.	*pluvi-alis*	‘rainy’
		leg-alis	‘legal’

5.3.1 Setup

The data comes from the Perseus project (Smith *et al.*, 2000), which contains Old and Classical Latin texts from the 3rd century BCE through the 2nd century CE. Words are annotated for frequency. We extracted words containing two liquids and ending in *-alis*/*-aris*, and removed non-adjectives. The result is a dataset of 121 words. We treat /-aLis/ as the underlying form of [*-alis*]/[*-aris*], where /L/ is unspecified for [?lat]. We map orthographic segments directly to phonemes, treating <v> as the semivowel [w], <c> as [k], <x> as [ks], and <ll> as [l]. We also dropped the <h> from <th>, <kh>, and <ph>, treating <h> as marking aspiration. We list the segment features in Tab. 13 in the appendix.

The training data is an 80% frequency-weighted sample of words (97), and the testing data is the remaining 20% of words. This sampling was repeated 30 times, and models' performances are computed as averages over these 30 train/test splits.

5.3.2 Results

Again (Tab. 7), D2L is the most accurate model. D2L discovered rule (54), where /L/ takes the opposite value of [lat] from the consonant to its left on a consonant tier.

Table 7: Accuracy of models on held-out test words after learning Latin liquid dissimilation.

Model	Test Accuracy
D2L	**0.9708 ± 0.03**
GG	0.2708 ± 0.07
GR	0.8861 ± 0.12
LSTM	0.7444 ± 0.09
TSLIA	0.2708 ± 0.07
3G	0.9194 ± 0.04

(54) $\textsc{Disagree}([?\mathrm{lat}], \{\mathrm{lat}\}) / [+\mathrm{cons}]\ \_\_ \circ \mathrm{proj}(\cdot, [+\mathrm{cons}])$

We use features from Cser (2010), in which [l] is the only [+lat] segment. Thus, this rule in fact says that /L/ disharmonizes to [+lat] [l] when tier-adjacent to any consonant except [+lat] [l], for which it disharmonizes to [−lat] [r]. In effect then, [l] surfaces for /L/, except when tier-adjacent to a preceding [l], in which case it surfaces as [r]. Thus [r] and the [−cor] consonants, which block dissimilation according to Cser (2010), are preserved on the tier, consistent with Cser (2010)'s analysis. The [+cor] consonants {s, t, d, n} are also included on the tier. Cser (2010) notes that there is no data either way for whether [d] blocks. In our data, there is insufficient motivation for D2L to remove {s, t, n} from the tier.

6 Discussion

We intend D2L as a cognitive model for how humans learn phonological alternations, at roughly the *algorithmic level* in the sense of Marr (1982). That is, D2L constitutes a hypothesis that human learners roughly follow the steps laid out by D2L—tracking adjacent dependencies and iteratively discarding those that prevent prediction of the surface form. This paper constitutes a presentation of that hypothesis (§ 2) and independent motivation for it (§ 2.1), followed by initial supporting evidence on the grounds of prior artificial language studies (§ 4) and new computational modeling experiments (§ 5). As an explicit computational model, it can be used to generate predictions for further experimental studies.

We believe D2L is of importance to phonological theory because it demonstrates that phonological tiers can be learned from naturalistic data, even in the presence of exceptions, by tracking only adjacent dependencies, and that such representations arise naturally from a learner grounded in humans' proclivity for tracking adjacent dependencies (§ 2.1). Moreover, if D2L is on the right track regarding how humans generalize, then it also holds some explanatory power for why phonological structures are the way they are (Dresher, 1999; Heinz, 2009, 2010). In particular, because D2L is restricted to only tracking adjacent dependencies, the only option it has when it is unable to predict the surface form of an alternating segment is to delete the adjacent segments that are getting in the way and hope that the relevant dependencies will then be accessible adjacently on the new representation. Thus, the fact that a wide range of phonological phenomena can be described in terms of adjacent dependencies on some tier (Heinz, 2010; Jardine and Heinz, 2016; Burness and McMullin, 2021) could be a reflection of the fact that the human mind's attention is initially drawn to adjacent items, and the natural resulting learning algorithm produces tier-like representations as a byproduct.

6.1 D2L Compared to Other Models

We will highlight how D2L compares to other models, and why its functioning leads it to match human behavior in the artificial language experiments (§ 4) and succeed at learning natural language alternations (§ 5).

The artificial language experiments (§ 4) used poverty-of-stimulus paradigms, which precisely design the exposure (training) data so as to underdetermine what process underlies the observed alternation. Thus, when Finley (2011)'s exposure data contains CVS̲V-S̲V words, it could be that harmony applies strictly to sibilants across intervening vowels (i.e. S̲VS̲) or also across intervening non-sibilants (e.g. S̲VCVS̲V). The exposure data contains no words where sibilants are separated by more than a vowel, so the exposure data underdetermines which of these generalizations underlies the harmony. Similarly, when the exposure data contains S̲VCV-S̲V words, it could be that harmony applies strictly to sibilants where both consonants and vowels intervene (but not when *only* a vowel intervenes) or whenever anything other than a sibilant intervenes. Again, the choice is underdetermined. The same logic holds for McMullin and Hansson (2019)'s experiments.

As evidenced by the human behavior, something is critically different between the cases where the exposure data contains CVS̲V-S̲V words compared to when it contains S̲VCV-S̲V words. D2L's incremental deleting of adjacent dependencies leads it to delete only vowels (V) when exposed to CVS̲V-S̲V words, and both vowels and non-sibilant consonants (V and C) when exposed S̲VCV-S̲V words, mirroring the asymmetry observed in human behavior. In contrast, consider the way GG constructs a tier. When the exposure data is CVS̲V-S̲V words and S̲VCV-S̲V words, the Hayes and Wilson (2008) model, which GG uses, may observe that non-harmonizing S̲VS̲ sequences are conspicuously absent and learn the constraints *[s][][ʃ] and *[ʃ][][s]. When the exposure data is SVCV-SV words, it may notice that *any* (harmonizing or not) SVS sequences are conspicuously absent and learn the constraints *[s][][ʃ], *[s][][s], *[ʃ][][s], and *[ʃ][][ʃ].[10] In both cases, because GG uses the smallest natural class containing X and Y in *X[]Y constraints, it will project the smallest natural class containing {s, ʃ}, which is the [+stri] tier. Thus, fundamentally, GG is not capable of distinguishing the case where sibilants harmonize only across intervening vowels from the case where sibilants harmonize across both vowels and non-sibilant consonants.

We see the same situation when we turn to natural language data, where D2L is again able to handle blockers in Latin liquid dissimilation (§ 5.3), but GG, which may induce the constraint *[l][][l] if such sequences are sufficiently absent in the training data, induces the smallest natural class containing {l}, which would be [+lat], since [l] is the only lateral consonant. As a result, neither the [r] nor [−cor] blockers will be projected on the tier. In other words, GG is only sensitive to the harmonizing or dissimilating segments and thus cannot capture blockers unless they coincidentally fall in the smallest natural class subsuming X and Y in the *X[]Y constraints that Hayes and Wilson (2008)'s model extracts.

Goldsmith and Riggle (2012)'s model (GR), which was intended as an information-theoretic attempt to justify the use of a vowel tier in analyzing Finnish vowel harmony, used Goldsmith and Xanthos (2009)'s HMM to extract tiers. However, this approach is only well-suited for extracting the consonant and vowel tiers. This is because Goldsmith and Xanthos (2009)'s HMM is a state-transition model with two hidden states; it extracts the two classes of segments that maximizes the probability of the data. Intuitively, this is maximized when the two hidden categories transition sequentially in words. In such a case, the probability of transitioning from one state to the next will be high and the probability of staying in the current state will be low. In most imaginable linguistic cases, words tend to switch between consonants and vowels—for example CVCVCV and CVCCVC are more frequent than CCVV or CCCCVCC. This is hardly a categorical fact, but it is a robust statistical trend. Thus, the HMM's two hidden states tend to robustly correspond to the categories *consonant* and *vowel*. For this reason, GR is not able to flexibly extend to alternations involving tiers other than the [±cons] tiers.

The Jardine and McMullin (2017) model (TSLIA), as well as Burness and McMullin (2019, 2021)'s model, were proposed primarily for proving learnability results about tier-local generalizations. Such proofs require certain assumptions about the data available to the

[10] In practice, we found that the exposure dataset was small enough that GG did not always pick up on these regularities.

learner. These conditions specify a *characteristic sample*, which, as discussed by Jardine and McMullin (2017, p. 75), may not be satisfied in natural language data (e.g., due to interaction with other phonotactic constraints). TSLIA's performance suggests that such a sample is indeed not present in our datasets. Nevertheless, as discussed in § 3.2, these models share with D2L the process of iterative deletion of segments from the tier. Thus, D2L encodes the spirit of these models, while using featural representations to generalize over classes of segments, and the Tolerance Principle to deal with the sparsity and exceptions inevitable in naturalistic data.

The LSTM neural network model failed to match human behavior, as it generalized harmony/dissimilation to all types of test instances in the artificial language experiments, including those where harmony is blocked for humans by intervening segments. This suggests that LSTM may not be well-suited for capturing blockers. Moreover, LSTM appears to be overly sensitive to the baseline frequency of an alternating segment's surface forms. In Latin, the most frequent form of /L/ is [l], and 100% of LSTM's errors result from producing /L/ → *[l] instead of [r]. By far the most common error is producing the most frequent surface form of an alternating segment in Turkish and Finnish as well.

6.2 Future Directions

In its current form, D2L learns categorical rules. Extending D2L to handle variation is an important step for future research. This will likely involve the current model for constructing the structural description of rules, but would need to allow them to be probabilistic (Labov, 1969; Mayer, 2021).

In this paper, we considered non-interacting (dis)harmony processes. A model like the *Parsimonious Local Phonology* learner from Belth (2023a) provides a broader framework for how D2L could fit in a larger theory of phonological learning that includes epenthesis, deletion, and multiple generalizations.

Another direction for future research involves investigating whether metrical stress and tonal patterns can be learned by a similar process. For example, Jardine (2016a) argued that while most of segmental phonology can be characterized in terms of tier-locality, there are numerous processes in tonal phonology (e.g. tonal plateauing) that may require greater expressive power (unbounded circumambient) to account for. Moreover, McCollum *et al.* (2020) argued that Tutrugbu ATR harmony requires the same expressive power as Jardine (2016a)'s tonal analyses, suggesting that unbounded circumambient processes may be more prevalent in segmental phonology than previously thought. The key characteristic of unbounded circumambient phenomena is the involvement of dependencies that are arbitrarily far away in *both directions*. Thus, future research could investigate whether D2L's approach of iteratively deleting adjacent, unuseful, segments could render local the relevant dependencies from both directions.

References

Akiba, Takuya, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2623–2631. ACM.

Aksënova, Alëna. 2020. Sigmapie. https://github.com/alenaks/SigmaPie.

Albright, Adam, and Bruce Hayes. 2002. Modeling english past tense intuitions with minimal generalization. In *Proceedings of the ACL-02 workshop on Morphological and phonological learning*, 58–69.

Albright, Adam, and Bruce Hayes. 2003. Rules vs. analogy in english past tenses: A computational/experimental study. *Cognition* 90:119–161.

Altan, Asli. 2009. Acquisition of vowel harmony in Turkish. *Dilbilim 35. Yıl Yazıları* 9–26.

Andersson, Samuel, Hossep Dolatian, and Yiding Hao. 2020. Computing vowel harmony: The generative capacity of search & copy. In *Proceedings of the Annual Meetings on Phonology*, vol. 7.

Aslin, Richard N, Jenny R Saffran, and Elissa L Newport. 1998. Computation of conditional probability statistics by 8-month-old infants. *Psychological science* 9:321–324.

Baer-Henney, Dinah, and Ruben van de Vijver. 2012. On the role of substance, locality, and amount of exposure in the acquisition of morphophonemic alternations. *Laboratory Phonology* 3:221–249.

Belth, Caleb. 2023a. A learning-based account of local phonological processes. *Phonology* In Press.

Belth, Caleb. 2023b. A learning-based account of non-productivity in Dutch voicing alternations. *Proceedings of the 48nd annual Boston University Conference on Language Development.* In Press.

Belth, Caleb. 2023c. Towards a learning-based account of underlying forms: A case study in Turkish. *Proceedings of the Society for Computation in Linguistics* .

Belth, Caleb, Sarah Payne, Deniz Beser, Jordan Kodner, and Charles Yang. 2021. The greedy and recursive search for morphological productivity. In *CogSci*.

Bennett, William G. 2013. Dissimilation, consonant harmony, and surface correspondence. Doctoral dissertation, Rutgers University.

Burness, Phillip, and Kevin McMullin. 2019. Efficient learning of output tier-based strictly 2-local functions. In *Proceedings of the 16th Meeting on the Mathematics of Language*, 78–90. Toronto, Canada: Association for Computational Linguistics.

Burness, Phillip, and Kevin McMullin. 2021. More efficiently identifying the tiers of strictly 2-local tier-based functions. In *Proceedings of the 17th Meeting on the Mathematics of Language*, 38–49. Umeå, Sweden: Association for Computational Linguistics.

Burness, Phillip, Kevin McMullin, and Jane Chandlee. 2021. Long-distance phonological processes as tier-based strictly local functions. *Glossa: a journal of general linguistics* 6.

Chomsky, Noam. 2001a. *Beyond explanatory adequacy*. Cambridge, MA: MIT Working Papers in Linguistics.

Chomsky, Noam. 2001b. Derivation by phase. In *Ken Hale: A life in language*, edited by Michael Kenstowicz, 1–52. Cambridge, MA: MIT Press.

Chomsky, Noam, and Morris Halle. 1968. *The sound pattern of English.*. Cambridge, MA: Harper & Row.

Clements, George N. 1976. *The autosegmental treatment of vowel harmony*. Indiana University Linguistics Club.

Clements, George N. 1980. *Vowel harmony in nonlinear generative phonology*. Indiana University Linguistics Club Bloomington.

Çöltekin, Çagri. 2010. A freely available morphological analyzer for turkish. In *LREC*, vol. 2, 19–28.
URL https://coltekin.net/cagri/papers/coltekin2010.pdf

Cotterell, Ryan, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The sigmorphon 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, 10–22.

Cotterell, Ryan, Nanyun Peng, and Jason Eisner. 2015. Modeling word forms using latent underlying morphs and phonology. *Transactions of the Association for Computational Linguistics* 3:433–447.

Cser, András. 2010. The -alis/-aris allomorphy revisited. In *Variation and change in morphology: selected papers from the 13th international morphology meeting*, 33–52. John Benjamins Publishing.

Dobrovolsky, Michael. 1982. Some thoughts on turkish voicing assimilation. In *Calgary Working Papers in Linguistics*, vol. 7, 1–6.

Dresher, Bezalel Elan. 1999. Charting the learning path: Cues to parameter setting. *Linguistic inquiry* 30:27–67.

Ellis, Kevin, Adam Albright, Armando Solar-Lezama, Joshua B Tenenbaum, and Timothy J O'Donnell. 2022. Synthesizing theories of human language with bayesian program induction. *Nature communications* 13:5024.
URL https://www.nature.com/articles/s41467-022-32012-w

Emond, Emeryse, and Rushen Shi. 2021. Infants' rule generalization is governed by the Tolerance Principle. In *Proceedings of the 45nd annual Boston University Conference on Language Development*, edited by Danielle Dionne and Lee-Ann Vidal Covas, 191–204.

Finley, Sara. 2011. The privileged status of locality in consonant harmony. *Journal of memory and language* 65:74–83.

Fiser, József, and Richard N Aslin. 2002. Statistical learning of higher-order temporal structure from visual shape sequences. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 28:458.

Frédéric, Mailhot, and Charles Reiss. 2007. Computing long-distance dependencies in vowel harmony. *Biolinguistics* 1:28–48.

Gold, E Mark. 1967. Language identification in the limit. *Information and control* 10:447–474.

Goldsmith, John. 1976. Autosegmental phonology. Doctoral dissertation, Massachusetts Institute of Technology.

Goldsmith, John, and Jason Riggle. 2012. Information theoretic approaches to phonological structure: the case of finnish vowel harmony. *Natural Language & Linguistic Theory* 30:859–896.

Goldsmith, John, and Aris Xanthos. 2009. Learning phonological categories. *Language* 85:4–38.

Gómez, Rebecca, and Jessica Maye. 2005. The developmental trajectory of nonadjacent dependency learning. *Infancy* 7:183–206.

Gómez, Rebecca L. 2002. Variability and detection of invariant structure. *Psychological Science* 13:431–436.

Gouskova, Maria, and Gillian Gallagher. 2020. Inducing nonlocal constraints from baseline phonotactics. *Natural Language & Linguistic Theory* 38:77–116.

Graf, Thomas, and Nazila Shafiei. 2019. C-command dependencies as TSL string constraints. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*, 205–215.

Hare, Mary. 1990. The role of similarity in hungarian vowel harmony: a connectionist account. *Connection Science* 2:123–150.

Hayes, Bruce, and Colin Wilson. 2008. A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry* 39:379–440.

Hayward, Richard J. 1990. Notes on the Aari language. *Omotic language studies* 425–493.

Heinz, Jeffrey. 2009. On the role of locality in learning stress patterns. *Phonology* 26:303–351.

Heinz, Jeffrey. 2010. Learning long-distance phonotactics. *Linguistic Inquiry* 41:623–661.

Heinz, Jeffrey, Chetan Rawal, and Herbert G. Tanner. 2011. Tier-based strictly local constraints for phonology. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 58–64. USA: Association for Computational Linguistics.

Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9:1735–1780.

Hua, Wenyue, Adam Jardine, and Huteng Dai. 2020. Learning underlying representations and input-strictly-local functions. In *Proceedings of the 37th West Coast Conference on Formal Linguistics*.

Van der Hulst, Harry. 2016. Vowel harmony. In *Oxford Research Encyclopedia of Linguistics*. Oxford University Press.

Jardine, Adam. 2016a. Computationally, tone is different. *Phonology* 33:247–283.

Jardine, Adam. 2016b. Learning tiers for long-distance phonotactics. In *Proceedings of the 6th conference on Generative Approaches to Language Acquisition North America (GALANA)*, 60–72.

Jardine, Adam, and Jeffrey Heinz. 2016. Learning tier-based strictly 2-local languages. *Transactions of the Association for Computational Linguistics* 4:87–98.

Jardine, Adam, and Kevin McMullin. 2017. Efficient learning of tier-based strictly k-local languages. In *Language and Automata Theory and Applications*, edited by Frank Drewes, Carlos Martín-Vide, and Bianca Truthe, 64–76. Springer.

Kabak, Bariş. 2011. *Turkish Vowel Harmony*, chap. 118, 1–24. John Wiley & Sons, Ltd.

Kingma, Diederik P., and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, edited by Yoshua Bengio and Yann LeCun.

Kiparsky, Paul. 1968. *How abstract is phonology?*. Indiana University Linguistics Club.

Kirov, Christo, and Ryan Cotterell. 2018. Recurrent neural networks in linguistic theory: Revisiting pinker and prince (1988) and the past tense debate. *Transactions of the Association for Computational Linguistics* 6:651–665.

Korn, David. 1969. Types of labial vowel harmony in the Turkic languages. *Anthropological Linguistics* 11:98–106.

Kornfilt, Jaklin. 2013. *Turkish*. Routledge.

Koulaguina, Elena, and Rushen Shi. 2019. Rule generalization from inconsistent input in early infancy. *Language Acquisition* 26:416–435.

Kurimo, Mikko, Sami Virpioja, Ville Turunen, and Krista Lagus. 2010. Morpho challenge 2005-2010: Evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, 87–95. ACL.

Labov, William. 1969. Contraction, deletion, and inherent variability of the English copula. *Language* 45:715–762.

Lambert, Dakotah. 2021. Grammar interpretations and learning tsl online. In *International Conference on Grammatical Inference*, 81–91. PMLR.

Lindblad, Vern M. 1990. *Neutralization in Uyghur*. University of Washington.

Marr, David. 1982. *Vision: A computational investigation into the human representation and processing of visual information*. San Francisco: W. H. Freeman and Company.

Mayer, Connor. 2021. Capturing gradience in long-distance phonology using probabilistic tier-based strictly local grammars. *Proceedings of the Society for Computation in Linguistics* 4:39–50.

McCollum, Adam G., Eric Baković, Anna Mai, and Eric Meinhardt. 2020. Unbounded circumambient patterns in segmental phonology. *Phonology* 37:215–255.

McCurdy, Kate, Sharon Goldwater, and Adam Lopez. 2020. Inflecting when there's no majority: Limitations of encoder-decoder neural networks as cognitive models for German plurals. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1745–1756. Association for Computational Linguistics.

McMullin, Kevin, and Gunnar Ólafur Hansson. 2016. Long-distance phonotactics as tier-based strictly 2-local languages. In *Proceedings of the Annual Meeting on Phonology*.

McMullin, Kevin, and Gunnar Ólafur Hansson. 2019. Inductive learning of locality relations in segmental phonology. *Laboratory Phonology* 10.

McMullin, Kevin James. 2016. Tier-based locality in long-distance phonotactics: learnability and typology. Doctoral dissertation, University of British Columbia.

Nevins, Andrew. 2005. Conditions on (dis) harmony. Doctoral dissertation, Massachusetts Institute of Technology.

Nevins, Andrew. 2010. *Locality in vowel harmony*, vol. 55. Cambridge: MIT Press.

Newport, Elissa L, and Richard N Aslin. 2004. Learning at a distance I. statistical learning of non-adjacent dependencies. *Cognitive psychology* 48:127–162.

O'Hara, Charlie. 2017. How abstract is more abstract? learning abstract underlying representations. *Phonology* 34:325–345.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, 8024–8035. Curran Associates, Inc.

Peperkamp, Sharon, Rozenn Le Calvez, Jean-Pierre Nadal, and Emmanuel Dupoux. 2006. The acquisition of allophonic rules: Statistical learning with linguistic constraints. *Cognition* 101:B31–B42.

Richter, Caitlin. 2018. Learning allophones: What input is necessary. In *Proceedings of the 42nd annual Boston University Conference on Language Development.*. Cascadilla Press.

Richter, Caitlin. 2021. Alternation-sensitive phoneme learning: Implications for children's development and language change. Doctoral dissertation, University of Pennsylvania.

Ringe, Don, and Joseph F Eska. 2013. *Historical linguistics: Toward a twenty-first century reintegration*. Cambridge University Press.

Ringen, Catherine O, and Orvokki Heinämäki. 1999. Variation in Finnish vowel harmony: An OT account. *Natural Language & Linguistic Theory* 17:303–337.

Rodd, Jennifer. 1997. Recurrent neural-network learning of phonological regularities in Turkish. In *Computational Natural Language Learning (CoNLL)*.

Rose, Sharon, and Rachel Walker. 2004. A typology of consonant agreement as correspondence. *Language* 80:475–531.

Saffran, Jenny R, Richard N Aslin, and Elissa L Newport. 1996. Statistical learning by 8-month-old infants. *Science* 274:1926–1928.

Saffran, Jenny R, Elizabeth K Johnson, Richard N Aslin, and Elissa L Newport. 1999. Statistical learning of tone sequences by human infants and adults. *Cognition* 70:27–52.

Saffran, Jenny R., Elissa L. Newport, Richard N. Aslin, Rachel A. Tunick, and Sandra Barrueco. 1997. Incidental language learning: Listening (and learning) out of the corner of your ear. *Psychological Science* 8:101–105.

Samuels, Bridget D. 2009a. Structure & specification in harmony. In *North East Linguistics Society (NELS)*.

Samuels, Bridget D. 2009b. The structure of phonological theory. Doctoral dissertation, Harvard University.

Santelmann, Lynn, and Peter W. Jusczyk. 1998. Sensitivity to discontinuous dependencies in language learners: Evidence for limitations in processing space. *Cognition* 69:105–134.

Schuler, Kathryn, Charles Yang, and Elissa Newport. 2016. Testing the Tolerance Principle: Children form productive rules when it is more computationally efficient to do so. In *CogSci*.

Smith, Caitlin, Charlie O'Hara, Eric Rosen, and Paul Smolensky. 2021. Emergent gestural scores in a recurrent neural network model of vowel harmony. *Proceedings of the Society for Computation in Linguistics* 4:61–70.

Smith, David A, Jeffrey A Rydberg-Cox, and Gregory R Crane. 2000. The Perseus Project: A digital library for the humanities. *Literary and Linguistic Computing* 15:15–25.

Suomi, Kari, Juhani Toivanen, and Riikka Ylitalo. 2008. *Finnish Sound Structure: Phonetics, Phonology, Phonotactics and Prosody*, vol. 9 of *Studia Humaniora Ouluensia*. University of Oulu.

Tesar, Bruce. 2013. *Output-Driven Phonology: Theory and Learning*. Cambridge: Cambridge University Press.

White, James, René Kager, Tal Linzen, Giorgos Markopoulos, Alexander Martin, Andrew Nevins, Sharon Peperkamp, Krisztina Polgárdi, Nina Topintzi, and Ruben van De Vijver. 2018. Preference for locality is affected by the prefix/suffix asymmetry: Evidence from artificial language learning. In *NELS*, 207–220.

Yang, Charles. 2016. *The price of linguistic productivity: How children learn to break the rules of language*. Cambridge: MIT Press.

A Theoretical Connections

A.1 Relationship to Search and Copy

A number of search-based accounts of vowel harmony have been proposed (Nevins, 2005; Frédéric and Reiss, 2007; Samuels, 2009a,b; Nevins, 2010; Andersson *et al.*, 2020), in which underspecified vowels trigger a linear search for a valid donor from which to copy a feature value for any unspecified feature(s); the search skips invalid donors. Search-and-copy accounts do not provide an account of how the set of segments skipped during the search is learned, which D2L provides via the deletion set. In a tier-projection account, a tier representation is explicitly projected, which renders the relevant dependency *adjacent*. In contrast, search-and-copy operates over an input sequence, skipping the 'non-tier' elements but not projecting an explicit tier. Our reason for treating the tier as a projection is the experimental evidence that adjacency is cognitively prominent (see discussion in § 1).

A.2 Relationship to Syntactic Agree

Search-and-Copy approaches recognize the similarity to syntactic Agree (Chomsky, 2001b,a). In both, a recipient searches for a valid donor to specify an unspecified feature value, and take their value from the closest valid donor (we refer readers to Nevins 2010 for extensive discussion). Formal-language-theoretic work has also recognized the relationship between tier-based phonological generalizations and syntactic dependencies, and begun attempts to formalize the relationship (Graf and Shafiei, 2019).

The notion of closeness differs in syntax and phonology (c-command vs. string proximity), but this may be a reflection of the data structure over which dependencies are computed, rather than the dependency-inducing operation itself. In our view, the conspicuous similarities between the two phenomena suggest that the same cognitive mechanism may be at play in both. Of course, this view is highly conjectural and needs much more investigation, but we view it as a promising line of inquiry, and adopted the terminology Agree and Disagree to emphasize this connection.

A.3 LSTM Details

We use a Long Short-Term Memory (LSTM; Hochreiter and Schmidhuber 1997) as our RNN architecture because it is better-suited for capturing long-distance dependencies than the older models used in phonology (Hare, 1990; Rodd, 1997). We used a Pytorch (Paszke *et al.*, 2019) implementation. Each input sequence is first run through an embedding layer with learned weights, which maps each segment to a real-valued vector representation (embedding) of dimension $d \in \mathbb{N}$. These embeddings are then run through a bidirectional LSTM, which yields a real-valued hidden representation of dimension $h \in \mathbb{N}$ for each embedding. By using a bidirectional LSTM (as opposed to a unidirectional LSTM), the model is capable of capturing dependencies to both the left and the right of an alternating segment, which is important for modeling either leftward- or rightward-spreading processes. Each hidden representation is then mapped to the output alphabet (i.e. the set of possible surface forms for the underlying, input segment) by a fully-connected layer, and a softmax converts the scores to a distribution over the output alphabet. We use a cross-entropy loss, which encourages—via backpropagating the error—the model to assign high probability ($\approx$ 1) to the correct surface form and low probability ($\approx$ 0) to all other possible surface forms. Since we wish to train the model only to predict the surface form of each underlyingly underspecified segments, the cross-entropy loss ignores the model's predictions for underlyingly specified segments. As noted in the text, this makes the problem strictly *easier* for the LSTM than requiring that it predict the *entire* surface sequence. We make this simplification because we think it makes for a more fair comparison to D2L, which also has access to the underlying forms, from which it can derive which segments are alternating.

We train the model on each dataset using the Adam optimizer (Kingma and Ba, 2015) and perform hyperparameter tuning with the Optuna Python package (Akiba *et al.*, 2019). We used the Tree-structured Parzen Estimator (TPE) algorithm, which improves hyperparameter search compared to simpler methods like grid or random search. For hyperparameter tuning, we trained a model with each hyperparameter combination over an 80% sample of the training data, using the remaining 20% as validation data. We carried out the TPE algorithm for 30 iterations, which allows for 30 combinations of hyperparameters to be tested. After these 30 iterations, we used the hyperparameters with the smallest validation loss to train a final model on the entire training data. Hyperparameters were the embedding dimension $d \in \{16, 32, 64, 128, 256, 512\}$, hidden dimension $h \in \{16, 32, 64, 128, 256, 512\}$, learning rate in [0.0001, 0.1], and number of epochs in $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$.

Table 8: Features for comparison to Finley (2011).

SEG	cons	voice	son	nas	lab	cor	ant	strid	back	round	hi	low
S	+	−	−	−	−	+		+				
a	−	+	+						−	−	−	+
b	+	+	−	−	+	−	+	−				
d	+	+	−	−	−	+	+	−				
e	−	+	+						−	−	−	−
g	+	+	−	−	−	−	−	−				
i	−	+	+						−	−	+	−
k	+	−	−	−	−	−	−	−				
m	+	+	+	+	+	−	+	−				
n	+	+	+	+	−	+	+	−				
o	−	+	+						+	+	−	−
p	+	−	−	−	+	−	+	−				
s	+	−	−	−	−	+	+	+				
t	+	−	−	−	−	+	+	−				
u	−	+	+						+	+	+	−
ʃ	+	−	−	−	−	+	−	+				

Table 9: Features for comparison to McMullin and Hansson (2019).

SEG	cons	voice	cont	son	nas	lab	cor	ant	strid	lat	back	hi
L	+	+	−	+	−	−			−			
b	+	+	−	−	−	+	−	+	−	−		
d	+	+	−	−	−	−	+	+	−	−		
e	−	+		+							−	−
g	+	+	−	−	−	−	−	−	−	−		
i	−	+		+							−	+
k	+	−	−	−	−	−	−	−	−	−		
l	+	+	−	+	−	−	+	+	−	+		
m	+	+	−	+	+	+	−	+	−	−		
n	+	+	−	+	+	−	+	+	−	−		
o	−	+		+							+	−
p	+	−	−	−	−	+	−	+	−	−		
t	+	−	−	−	−	−	+	+	−	−		
u	−	+		+							+	+
ɹ	+	+	−	+	−	−	−	−	−	−		

B Feature Specifications

Table 10: Features for Turkish.

SEG	cons	voice	cont	son	nas	lab	cor	ant	strid	lat	back	round	hi
A	−	+		+								−	−
D	+		−	−	−	−	+	+	−	−			
H	−	+		+									+
b	+	+	−	−	−	+	−	+	−	−			
d	+	+	−	−	−	−	+	+	−	−			
e	−	+		+							−	−	−
f	+	−	+	−	−	+	−	−	−	−			
g	+	+	−	−	−	−	−	−	−	−			
h	+	−	+	−	−	−	−	−	−	−			
i	−	+		+							−	−	+
j	+	+	−	+	−	−	−	−	−	−			
k	+	−	−	−	−	−	−	−	−	−			
l	+	+	−	+	−	−	+	+	−	+			
m	+	+	−	+	+	+	−	+	−	−			
n	+	+	−	+	+	−	+	+	−	−			
o	−	+		+							+	+	−
p	+	−	−	−	−	+	−	+	−	−			
r	+	+	−	+	−	−	+	−	−	−			
s	+	−	+	−	−	−	+	+	+	−			
t	+	−	−	−	−	−	+	+	−	−			
u	−	+		+							+	+	+
v	+	+	+	−	−	+	−	−	−	−			
w	+	+	−	+	−	+	−	−	−	−			
y	−	+		+							−	+	+

Table 11: Features for Turkish (continued).

SEG	cons	voice	cont	son	nas	sib	lab	cor	ant	strid	lat	back	round	hi
z	+	+	+	−	−	−	+	+	+	−				
ø	−	+		+							−	+	−	
ɑ	−	+		+							+	−	−	
ɯ	−	+		+							+	−	+	
ɰ	+	+	−	+	−	−	−	−	−	−				
ʃ	+	−	+	−	−	−	+	−	+	−				
ʒ	+	+	+	−	−	−	+	−	+	−				
d͡ʒ	+	+	−	−	−	−	+	−	+	−				
t͡ʃ	+	−	−	−	−	−	+	−	+	−				

Table 12: Features for Finnish.

SEG	cons	voice	cont	son	nas	lab	cor	ant	strid	lat	back	round	hi	low
A	−	+		+								−	−	+
O	−	+		+								+	−	−
U	−	+		+								+	+	−
b	+	+	−	−	−	+	−	+	−	−				
d	+	+	−	−	−	−	+	+	−	−				
e	−	+		+							−	−	−	−
f	+	−	+	−	−	+	−	−	−	−				
g	+	+	−	−	−	−	−	−	−	−				
h	+	−	+	−	−	−	−	−	−	−				
i	−	+		+							−	−	+	−
j	+	+	−	+	−	−	−	−	−	−				
k	+	−	−	−	−	−	−	−	−	−				
l	+	+	−	+	−	−	+	+	−	+				
m	+	+	−	+	+	+	−	+	−	−				
n	+	+	−	+	+	−	+	+	−	−				
o	−	+		+							+	+	−	−
p	+	−	−	−	−	+	−	+	−	−				
r	+	+	−	+	−	−	+	−	−	−				
s	+	−	+	−	−	−	+	+	+	−				
t	+	−	−	−	−	−	+	+	−	−				
u	−	+		+							+	+	+	−
v	+	+	+	−	−	+	−	−	−	−				
y	−	+		+							−	+	+	−
æ	−	+		+							−	−	−	+
ø	−	+		+							−	+	−	−
ɑ	−	+		+							+	−	−	+

Table 13: Features for Latin. The consonant features are taken from Cser (2010).

SEG	cons	cor	lab	voice	son	nas	con	lat	high	back	round	hi	low
L	+	+	−	+	+	−	+		−				
a	−			+	+					−	−	−	+
b	+	−	+	+	−	−	−	−	−				
d	+	+	−	+	−	−	−	−	−				
e	−			+	+					−	−	−	−
f	+	−	+	−	−	−	+	−	−				
g	+	−	−	+	−	−	−	−	+				
h	+	−	−	+	−	−	+	−	−				
i	−			+	+					−	−	+	−
k	+	−	−	−	−	−	−	−	+				
l	+	+	−	+	+	−	+	+	−				
m	+	−	+	+	+	+	−	−	−				
n	+	+	−	+	+	+	−	−	−				
o	−			+	+					+	+	−	−
p	+	−	+	−	−	−	−	−	−				
r	+	+	−	+	+	−	+	−	−				
s	+	+	−	−	−	−	+	−	−				
t	+	+	−	−	−	−	−	−	−				
u	−			+	+					+	+	+	−
w	+	−	+	+	+	−	+	−	+				