Korean Journal of English Language and Linguistics, Vol 23, October 2023, pp. 839-858
DOI: 10.15738/kjell.23..202310.839

KOREAN JOURNAL OF ENGLISH LANGUAGE AND LINGUISTICS

ISSN: 1598-1398 / e-ISSN 2586-7474

http://journal.kasell.or.kr

Learning the distribution of English *-al* and *-ar* suffixes using deep neural networks

Hyesun Cho (Dankook University)

Received: August 5, 2023
Revised: October 1, 2023
Accepted: October 15, 2023

Hyesun Cho
Associate Professor,
Department of Education,
Graduate School of Education,
Dankook University
Email: hscho@dankook.ac.kr

ABSTRACT

Cho, Hyesun. 2023. Learning the distribution of English *-al* and *-ar* suffixes using deep neural networks. *Korean Journal of English Language and Linguistics* 23, 839-858.

This study utilized an ensemble of recurrent and convolution neural networks, labeled deep neural networks (DNN) to learn the variable distribution of English suffixes *-al* and *-ar*. The DNN predictions were compared against the predictions of the maximum entropy phonotactic learner (PL). An examination of 1,479 adjectives suffixed with *-al* and *-ar* revealed that the suffix and the stem-final segment always underwent liquid dissimilation if the stem-final segment was a liquid (e.g. *so<u>l</u>ar*, *plu<u>r</u>al*). The suffix was *-ar* if the stem-final segment was /l/; conversely, the suffix *-al* occurred after /r/. The suffixes were found to vary if the stem-final segment was not liquid (e.g. *lo<u>c</u>al*, *lu<u>n</u>ar*). The learning results revealed that the DNN exhibited higher classification accuracy (97.3%) than the PL (89.4%). The PL assigned higher or equal probabilities to unattested word forms than to attested ones in 10.5% of the test data. The DNN successfully learned the variable distribution patterns of the suffixes observed in the training data. The probability of the suffix *-al* being predicted by the DNN also effectively showed the gradual distance effects of liquids on liquid dissimilation and segmental blocking. The DNN model learned the sigmoid curve commonly observed in linguistic data.

KEYWORDS

English suffix, deep neural networks, liquid dissimilation, lateral dissimilation, classification

1. Introduction

1.1 English Suffixes *-al* and *-ar*

The English suffixes *-al* and *-ar* are derivational suffixes forming adjectives from nouns (*tonal* /toʊn+əl/ [toʊn-əl], *polar* /poʊl+əl/ [poʊl-ər]) or bound roots (*central* /sɛntr+əl/ [sɛntr-əl]). The origin of *-al* and *-ar* was the Latinate adjectival suffix /-ālis/ (/nāv+ālis/ → [nāv-ālis] 'of ships'). /-ālis/ surfaces as [-āris] when the base contains an /l/ (/sōl+ālis/ → [sōl-āris] 'of the sun, solar'). Words in (1) show adjectives formed with *-al* (1a) and adjectives formed with *-ar* (1b, c) (Zymet 2015, Stanton 2017, Carstairs-McCarthy 2018, and from the data collected in the present study).

(1) a. tonal, habitual, distal, apical (No /l/ in the stem)
b. solar, polar, regular, circular (Stem-final [l])
c. lunar, linear, plantar, bulbar (Stem-internal [l])

When no /l/ is in the stem, the suffix *-al* attaches, as in (1a). When the stem ends in an /l/, as in (1b), or the stem contains an /l/, as in (1c), the suffix *-ar* attaches instead of *-al*. In brief, when the stem has an /l/, *-ar* attaches, and *-al* elsewhere. This is known as [+lateral] dissimilation or liquid dissimilation (Dressler 1971, Bennet 2015, Zymet 2015, Stanton 2017). More than one lateral is avoided within a word in the suffixation process of *-al*. The underlying form of the suffix is *-al* [-əl] because it appears in an environment where there is no liquid, as in (1a). The suffix surfaces as *-ar* [ər] to avoid having two laterals in the same word. The dissimilation environment can be local; an /l/ is avoided in the same syllable as the suffix, as in (1b) (*solar* /soʊl+əl/ [soʊl-ər]). It can also be non-local (long-distance); an /l/ is avoided even if it does not belong to the same syllable as the suffix, as in (1c) (*lunar* /lun+əl/ [lun-er]) (Zymet 2015:74). However, this rule of dissimilation is not always clear-cut, given the forms in (2).

(2) a. floral, plural (Segmental blocking)
b. local, loyal, skeletal (Lexical variation)

The stems in (2a) contain an /l/ (plural), but *-al* is chosen over *-ar*. In this case, an /r/ is in the stem-final position, which intervenes between the two /l/s (plural). The stem-final /r/ blocks the dissimilation of the two laterals, allowing *-al* to attach. This is known as segmental blocking in dissimilation (Bennet 2015, Stanton 2017). The stem-final /r/ is opaque in liquid dissimilation (Zymet 2015). However, the suffixation of *-al* in (2b) remains unexplained under this account because the stem-final segment is not /r/, and the stem contains an /l/, which is an environment where *-ar* would be expected. Thus, dissimilation is subject to lexical variation if the stem-final segment is not a liquid (*lunar* but *local*) (Zymet 2015). Different suffixes may attach even in the same phonological environment (*linear* [linɪər], *lineal* [linɪəl]). The choice of the suffix under this condition is thus unpredictable and probabilistic.

The [+lateral] dissimilation is to satisfy the constraint *[+lateral]...[+lateral], which penalizes the occurrence of two laterals within a form (Kenstowicz 1994, Stanton 2017). Similarly, the occurrence of *-al* in a plural form is to satisfy the constraint *[+rhotic]...[+rhotic], ranked above *[+lateral]...[+lateral]. However, these constraints cannot explain the occurrence of *-al* in words in (2b), where no /r/ is present between two laterals. Thus, we can

hypothesize that lateral dissimilation has a stochastic property when the stem-final segment is not a liquid.

Zymet (2015) showed that liquid segments in the stem have 'distance-based decay', where liquid dissimilation is more likely to occur if a liquid is closer to the suffix. A liquid farther from the suffix is less likely to trigger liquid dissimilation. When a liquid is in the same syllable as the suffix, the proportion of dissimilated forms is 99%. When a liquid is in the syllable right next to the suffix, the proportion of dissimilated forms is 65%. When there is one intervening syllable between liquid and suffix, the proportion drops to 10%. The probability of dissimilation exponentially decays as a function of the number of syllables to the suffix. The decay follows an exponential decay function ($d(x)=1/x^k$), where x is the distance to the suffix.

The choice of the suffix thus depends on the distance to the suffix and lexical variation, so it can be considered probabilistic. The probabilistic nature of the suffix choice can be best modeled using stochastic, probabilistic models. In this paper, we collect a list of *-al*/*-ar* suffixed adjectives in English and examine whether a deep neural-network model can learn the distribution of the suffixes and make correct predictions on the suffix. The model's results are compared with a baseline model, the maximum entropy model of phonotactics or the Phonotactic Learner (Hayes and Wilson 2008). The Phonotactic Learner was chosen because both models learn phonotactic patterns from large data and make probabilistic predictions on phonological forms.

1.2 Models for Phonotactic Classification

This paper will apply two models, the Phonotactic Learner by Hayes and Wilson (2008) ('PL') and an ensemble of two types of deep neural networks to learn the distribution of *-al* and *-ar*. This section describes each model and the nature of the task.

1.2.1 The Phonotactic Learner (Hayes and Wilson 2008)

The Phonotactic Learner (Hayes and Wilson 2008) is a model of phonotactics and phonotactic learning. The model learns a phonotactic grammar consisting of constraints and their weights from a set of phonological forms by a machine-learning method. Unlike most phonological models using OT constraints (e.g. Goldsmith and Johnson 2003, Potts et al. 2010), to run the PL, human linguists do not have to handcraft constraints for their data, but the entire set of constraints are inductively learned based on the surface forms in the training data. The constraints are formed from the possible combinations of the natural classes based on the feature specifications for the phonemes used in the training data. The model searches for a set of constraints and constraint weights that maximize the probability of the observed data using the criterion of the O/E ratio. The constraints that satisfy the O/E accuracy criteria are added to the phonotactics grammar for the given training data. The constraint weights are learned by stochastic gradient ascent (Della Pietra et al. 1997). The obtained grammar consists of constraints and their weights. The well-formedness of a phonological form is evaluated by summing the weights of the constraint violations by the form ('score'), as in Harmonic Grammar. The higher the score, the more ill-formed the form. The score is converted to a maxent value, and the probability of a form is its maxent value over the sum of the maxent values of all the forms in the data.

The Phonotactic Learner takes an inductive baseline approach. Their inductive baseline was the SPE-style linear feature bundle approach, to which they added autosegmental tiers (Goldsmith 1979) and metrical grids. The PL is often employed as a baseline model in phonological studies (Albright 2009, Daland et al. 2011) and as a baseline for comparison with neural networks for phonotactic learning (Mayer and Nelson 2020).

1.2.2 Deep learning language models

Computational language models are statistical learners trained to predict how likely an item is given a context. Neural networks have served as language models in that sense. With the advancement of computational capacity and technologies in deep neural networks, language models that use neural networks have been used for linguistic research. Neural networks have been gaining more attention in linguistics. Pater (2019) calls for a greater fusion between linguistic research and neural studies because both fields can benefit each other. So far, neural networks have been applied mostly in the field of syntax (e.g. Linzen 2019, Mahowald 2023, Wilcox et al. 2022), semantics (Petersen and Potts 2023), and computational linguistics to process natural language sentences and their meanings (Howard and Ruder 2018, Kim 2014, Mikolov et al. 2013, Vaswani et al. 2017). Linzen (2019), in response to Pater (2019), suggested ways to employ neural networks in linguistic research. According to Linzen (2019), the language models that use the RNN (recurrent neural networks) perform better than baseline models in various sentence acceptability tasks. Linzen (2019) suggested that a controlled experimental approach, rather than using naturally occurring sentences, will be necessary to successfully apply neural networks in linguistics research.

Research has focused on whether neural networks can learn specific structures in natural languages. Wilcox et al. (2022) showed that deep neural networks can learn filler-gap dependency and island constraints. Mahowald (2023) reports that GPT-3 gave acceptability judgments for the article + adjective + numeral + noun construction phrase (e.g. “a beautiful five trees”) similarly to human ratings. Petersen and Potts (2023) used large neural language models (LLMs) in the analysis of a wide range of word senses of English word break, arguing that “LLMs are powerful devices for studying lexical semantics in ways that can deeply inform linguistic theory”.

On the other hand, the application of neural networks has been limited in the field of phonology and phonetics compared to other areas. A few of these studies include Mayer and Nelson (2020), Mirea and Bicknell (2019), and Beguš (2020). Mayer and Nelson (2020) used a simple RNN language model for phonotactics learning in various languages, including the learning of English complex onset phonotactics. The RNN language model performed better than the baseline model (the PL). Mirea and Bicknell (2019) used the LSTM-RNN to test whether distinctive features are necessary for successful phonotactics learning. They found that the model with distinctive features and without them correlated well with human judgments, but the one without distinctive features performed better than the one with them. Beguš (2020) used GAN (Generative Adversarial Networks) for unsupervised learning of an allophonic distribution of aspirated voiceless stops. The model successfully learns the allophonic distribution of aspirated stops, generating conditional aspiration in the speech signal. Against this backdrop, the current work attempts to apply a deep neural network to study morpho-phonology, the distribution of English suffix *-al* and *-ar*.

Artificial neural networks with many hidden layers are called deep neural networks. The RNN (Recurrent Neural Network), especially Bi-LSTM-RNN (Bidirectional Long Short Term Memory RNN, Hochreiter and Schmidhuber 1997, Schuster and Paliwal 1997), is the most widely used in natural language processing because it can handle time-series sequences such as sentences. Unlike simple feed-forward neural networks, the RNN can use previous information, so it is useful when analyzing linguistic data, where contextual information is important (e.g. in Subject-Verb agreement). Another type of neural network, the CNN (Convolutional Neural Network), is commonly used in vision recognition. It has also been used in the field of text classification because it can effectively recognize collocational patterns in sentences (Kim 2014). Whereas the RNN learns the contextual information surrounding the words, the CNN can capture collocational patterns in text data. In the present study,

an ensemble model of the RNN (LSTM) and CNN (Kim 2021)[1] are used to classify stems that take *-al* or *-ar* suffixes. Ensemble is a machine-learning method that takes the average of several models for the best performance. The ensemble model (Kim 2021) used in this paper will be abbreviated as DNN (Deep Neural Network).

The PL and the DNN differ in their learning algorithms. The PL uses stochastic gradient ascent (Della Pietra et al. 1997) to find constraint weights that maximize the probability of all the words in the observed data. The DNN uses more advanced learning algorithms, such as backpropagation. From the viewpoint of linguists, a more meaningful difference between these models lies in the form of representations they process with. The PL uses OT-style constraints consisting of phonological features, while neural networks learn patterns directly from segment sequences without using phonological features or constraints. No linguistic representations, except segments, are assumed in the learning process of the DNN.

1.3 Phonotactics as a Classification Problem

Classification is a general and fundamental problem in machine learning (Domingos 1999, Cortes and Vapnik 1995, LeCun et al. 1998). Various types of data can be classified, including images (Krizhevsky et al. 2012) and text (McCallum and Nigam 1998, Joachims 1998, Kim 2014). Binary classification is classifying data into two predefined categories by assigning class labels using machine learning algorithms. For example, in email spam detection, an email is assigned spam or no spam labels depending on a classification algorithm. Multi-class classification is classifying examples into more than two categories, such as the iris data set (Fisher 1936)[2], consisting of three iris species types and their sepal and petal lengths and widths. Classification models can be tested to predict iris species accurately based on the variables.

A machine-learning algorithm that classifies data into certain categories is called a *classifier*, and there are various kinds of classifiers based on machine learning, such as random forests (Breiman 2001), support vector machines (Vapnik 1995), and multi-layer perceptron (Haykin 1994). The multi-layer perceptron (MLP) is a predecessor to deep neural networks, a neural network model with at least three hidden layers. Deep-learning models usually have more than three layers. Deep neural networks are also powerful classifiers for various kinds of data, such as images and text. Sentences can be classified depending on whether they are written by native or non-native speakers (Park et al. 2021, Cho 2021) or whether they convey positive or negative sentiments (Zhang et al. 2018).

Deep neural networks are an effective classifier because they can be trained to find the optimal regression line that distinguishes two classes that cannot be divided by a straight line (e.g. the XOR problem). After training, the networks can predict which class new examples belong to, along with their probabilities. The process includes learning weights through forward and backward propagation. The learned weights are updated by backward propagation. The weights are learned to minimize the errors (model's current prediction ($\hat{y}$) and real-world class (y) in the loss function, or the cross-entropy) (Goodfellow et al. 2015).

Classification is also a fundamental issue in linguistics. Linguistic theories hinge on categorizing linguistic properties (cf. Chomsky and Halle 1968). A central topic in linguistics depends on defining categories and analyzing their interactions. Knowledge of language is the knowledge of categorical differences between classes of linguistic elements. Linguistic categories are often binary (X or not X): for example, voicing ([+voice], [-

[1] https://github.com/kh-kim/simple-ntc
[2] https://archive.ics.uci.edu/ml/datasets/iris

voice]), tense ([+past], [-past]), and animacy [+animate]/[-animate]. Linguistic entities are represented by a combination of these binary categories, e.g. segments (/p/ [-voice, +labial, -sonorant], word meanings ('boy' [+human, -adult +male]). Phonemes (/p/,/b/,/t/, etc.) and syntactic categories (nouns, verbs, etc.) have multiple classes.

Choosing between two suffixes can also be treated as a classification problem. It is a task of classifying stems into two categories, *-al*-taking stems or *-ar*-taking stems. We recast the problem of choosing between suffixes as a classification task and use classifiers to see how well they classify phonological representations. In the present study, an ensemble of RNN and CNN is used to classify stems into their suffixes.

2.Method

2.1 Data Collection

Adjectives with the suffix *-al* are collected from a website (wordexample.com). From 12,661 adjectives ending in *-al*, the most frequent 1,000 adjectives were selected for the present study. Adjectives with the suffix *-ar* were not available from the same website, so they were collected from another website (wordmom.com), resulting in 405 adjectives ending in *-ar*[3]. These words were converted to pronunciation forms in ARPABET using Logios Lexicon Generation Tool[4]. The resulting pronunciation forms included multiple pronunciations for some words (e.g. *medical* [medɪkəl, medəkəl]), so the total number of adjectives in pronunciation forms was 1,102 (adjectives with *-al*) and 407 (adjectives with *-ar*). From here, monosyllabic words and compounds that do not contain the suffix (e.g. *real*, *in-ear*, *three-star*, *blue-collar*), obsolete forms (*militar*, *elementar*, *opacular*, *auxiliar*, *bilinguar*, *trilinguar*), and a non-existing word (*praemolar*) were excluded. This leaves 1,479 words (1,099 adjectives with *-al* and 380 adjectives with *-ar*).

These words were randomized and divided into training and testing data sets at an 8:2 ratio (training data: 1,184 words, test data: 295 words). The training data contained 879 *-al*-suffixed words and 305 *-ar*-suffixed words. The test data contained 220 *-al*-suffixed words and 75 *-ar*-suffixed words. The training data (1,184 words) and the test data (295 words) had the same ratios of stem-final and stem-internal segments (liquids, vowels, and non-liquid consonants).

2.2 Data Coding

The 1,479 words were coded with their suffixes (*-al*, *-ar*), stem-final segments (/l/, /r/, vowel, or non-liquid consonant), and stem-internal liquids (/l/, /r/, or no liquid). When there is more than one liquid in the stem-internal position (e.g. *infraclavicular* /..r..l..l/+/ar/), only the one immediately preceding the stem-final position is coded (/..l..l/+/ar/: stem-final is /l/, stem-internal is /l/), because those are known to be effective in the choice of suffixes in the previous literature (e.g. *plural*, *regular* in Stanton (2017)). Since the one closer to the stem final should be more important than the ones preceding it (Zymet 205), the liquids preceding the one immediately before the stem-final consonant were ignored in the present study (they existed in a total of 8 words (0.5%)).

[3] The *-ar*-suffixed adjectives were much less frequent than the *-al*-suffixed adjectives. Other data sources available online had less adjectives with *-ar*. For example, a wordlist from COCA (220,000 words, occurring at least 20 times in the corpus) contained 145 words ending in *-ar*, including non-English words such as *qsar*, *avsar* (https://www.wordfrequency.info/samples.asp).
[4] http://www.speech.cs.cmu.edu/tools/lextool.html

2.3 Learning with the Phonotactic Learner

To conduct learning simulations of the suffixes using the PL, we fed the model the learning (training) data, test data, and feature specifications of English phonemes. The maximum constraint length was set at 3 and the O/E ratio was 0.3. A projection tier for liquids ([+lateral] for /l/, [+rhotic] for /r/) is included to allow constraints to capture the occurrence patterns of /l/s and /r/s, ignoring other segments between them. According to the previous literature, vowels did not play an important role in the choice of the suffix, so only the consonants were left in the training and test sets. A part of the training data is shown in Figure 1. The test data had the same format.

```
V ZH W L
V T L
V K L
V K SH N L
W M Z K L
HH W M Z K L
Z N L
L Y L R
N M L K Y L R
B B L P L R
B G L N JH L R
```

Figure 1. Part of the training data for the PL
(The words are *visual*, *vital*, *vocal*, *vocational*, etc., from the top)

It should be noted that the PL is not a classifier. The model does not directly predict suffixes but only yields the well-formedness of test forms after learning from the training data. It does not classify stems into binary classes (-*al*-taking or -*ar*-taking). In the present study, the scores of attested and unattested forms were compared to determine the suffixes of test forms, and the ones that had a higher probability were considered as the suffix predicted for the forms. For this, the suffixes of the test data (295 attested forms) were reversed (-*al* to -*ar*, -*ar* to -*al*) (*fractal*-**fractar*, *popular*-**populal*), which is called 'unattested forms'. They were added to the test set; thus, the total number of the test words was 590 (295 attested forms, 295 unattested forms). It is assumed that the probabilities of the attested forms will be higher than those of the unattested forms. If the model assigns a higher probability to an attested form (e.g. *fractal*) than to an unattested form (e.g. **fractar*), it will be considered a correct prediction. If not, it will be considered an incorrect prediction. With these data and learner settings, the running time was 9 minutes 32 seconds.

2.4 Learning with Deep Neural Networks: An Ensemble of RNN(LSTM) and CNN

For our study, a neural-network text classifier by Kim (2021), an ensemble of the RNN and the CNN, was used for learning and predicting the suffixes. The PyTorch codes for the classification model were downloaded from the repository[5]. Figure 2 shows the sample of the training data. The training data consists of the label (suffix) column (left) and the stem column (right). Unlike the training data for the PL, it has stems and their associated suffixes separately. The DNN classifier learns the stems and their associated labels (suffixes) in the training set and predicts the suffixes for the stems in the test set. The test set has only the stem column without the suffix column. The stem is written in ARPABET, as generated by Logios Lexicon Generation Tool. Each stem is labeled with its associated suffix. The data contains consonants, as in the PL.

[5] https://github.com/kh-kim/simple-ntc


```
AL      M P R K
AL      N T M L R
AL      P R P Z SH N
AL      N G Y
AL      HH S T P TH L JH K
AL      JH Y K SH N
AL      S B
AR      S T B Y L
AR      K T NG G Y L
AR      D M N S Y L
AR      L Y L
```

Figure 2. Part of the training data for the DNN model. The first column is the labels (*-al*, *-ar* suffixes), and the second column is the stems. (Stems for words (from the top): *empirical*, *antimalarial*, *propositional*, etc.)

Note that training data for deep learning language models usually consist of sentences (Linzen et al. 2016, etc., but for word-level processing, see Sundermeyer et al. 2015) rather than words, as in here. Recurrent Neural Networks are typically used for sentence processing to compute probabilities of certain words in a sentence. On the other hand, each line of our training data is a stem consisting of phonemes. This way, deep-learning-based language models used for sentence processing can be applied to word-level processing. Phonemes in words correspond to words in sentences. Kim (2021)'s text classifier is designed for sentence classification, for example, classifying review comments on online shopping malls into negative or positive sentiments. Our study uses it for phonotactic classification by using the training data consisting of sequences of phonemes and their corresponding labels. The model computes the probability of a suffix given a stem.

The order of the lines in the training data was shuffled and divided into training and validation data at an 8:2 ratio (947, 237 each). Validation data was used during training to adjust parameters while avoiding overfitting. The running time to obtain the language model was 2 minutes and 6 seconds, with a CPU (Intel Core i7) on a MacBook Pro. Deep learning usually requires GPUs due to demanding computations, but in our data, the vocabulary (corresponding to each phoneme in our data) size was much smaller (only 27 phonemes used in the training data), and the length of each stem was shorter than usual sentence processing tasks, so the running time was short.

3. Results

3.1 Distribution of the Suffixes

The 1,479 words (1,099 with *-al*, 380 with *-ar*) described in Section 2.1 were coded with the type and presence of liquids in stem-final and stem-internal positions. Table 1 shows the distribution patterns, illustrating the example words according to their stem-final and stem-internal liquids and the number of words under each condition. Figure 1 is the plot that shows the frequency and proportion of suffixes presented in Table 1.

Table 1. Number of words with *-al*/*-ar* suffixes according to their stem-final and internal segments (N=1,479)

Stem-final	Stem-internal	*-al*		*-ar*	
/l/	/l/	-		mo**l**ecu**l**e-*ar*, **l**ocu**l**-*ar*	72 (5%)
	/r/	-		inte**r**ocu**l**-*ar*, **r**egu**l**-*ar*	92 (6%)
	No liquid	-		consu**l**-*ar*, simi**l**-*ar*	161 (11%)
/r/	/l/	f**l**o**r**-*al*, p**l**u**r**-*al*	27 (2%)	-	
	/r/	**r**u**r**-*al*, periphe**r**-*al*	23 (2%)	-	
	No liquid	doctor-*al*, inspi**r**-*al*	59 (4%)	-	
Vowel	/l/	**l**abi-*al*, li**n**e-*al*	22 (1%)	**l**ine-*ar*, conci**l**i-*ar*,	20 (1%)
	/r/	g**r**adu-*al*, mercu**r**i-*al*	113 (8%)	-	
	No liquid	hibitu-*al*, medi-*al*	43 (3%)	-	
Other	/l/	g**l**o**b**-*al*, lo**c**-*al*, fi**l**i-*al*	150 (10%)	bu**l**b-*ar*, p**l**an**t**-*ar*	32 (2%)
	/r/	i**r**oni**c**-*al*, ma**r**i**t**-*al*	335 (23%)	-	
	No liquid	conditio**n**-*al*, den**t**-*al*	327 (22%)	pal**m**-*ar*, the**n**-*ar*	3 (0%)

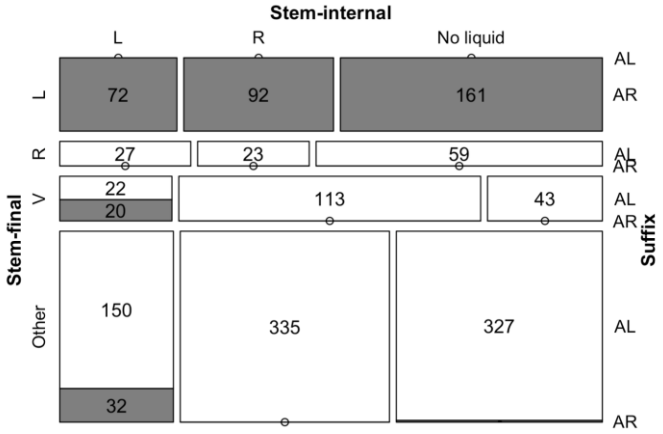


Figure 3. Number of words with *-al*/*-ar* suffixes according to their stem-final or internal liquids (white cells: *-al* suffixed words, grey cells: *-ar* suffixed words). Small circles indicate zero frequency.

Table 1 and Figure 3 clearly show that *-al* is never added after a stem-final /l/, and *-ar* is never added after a stem-final /r/. There is no adjective ending in /l-əl/, and /r-ər/. [+lateral] dissimilation is exceptionless if the stem-final segment is a liquid. The suffix is categorically predictable if the stem-final segment is /l/ or /r/, regardless of the stem-internal segments.

However, the suffix varies if the stem-final segment is not a liquid (non-liquid consonants and vowels). In that case, if there is a stem-internal /l/, the suffix *-al* and *-ar* are variable (*glob-al*, *bulb-ar*). If there is a stem-internal /r/, the suffix is always *-al* (*ironic-al*). If there is no liquid in the stem, *-al* is dominant (*condition-al*), with only

three words with *-ar* (*palm-ar* [pɑ́:mər][6] 'relating to the *palm'*, *then-ar* 'relating to the ball of the thumb', *hypothen-ar* 'relating to the hypothenar').

Among the vowel-final, /l/-internal stems, 22 take *-al* (e.g. *labial*) and 20 take *-ar* (e.g. *linear*). In 18 of these stems, an /l/ is the onset to the penultimate syllable (*filial* [fɪli-əl], *conciliar* [kənsili-ər]). Liquid segments in this position trigger liquid dissimilation, though not as strongly as stem-final liquids. According to Zymet (2015), 65% of the forms with a liquid in the penultimate syllable undergo liquid dissimilation. In our data, among the 18 vowel-final stems with an /l/ onset, 11 take *-ar*, undergoing liquid dissimilation (11/18=61%) (e.g. *conciliar*, *nuclear*), similar to Zymet's result. The other 7 stems did not undergo liquid dissimilation (e.g. *filial*, *loyal*). When /r/ is the onset to the stem-final vowel, suffixes are all *-al* (66/66=100%).

A chi-square test of independence showed that the relation between suffix and stem-final segments was significant ($\chi^2(3)=1209.1$, $p<.0001$). The relation between suffix and stem-internal liquids was also significant ($\chi^2(2)=54.26$, $p<.0001$). The distribution of the suffixes is significantly affected by stem-final and stem-internal liquids.

3.2 Learning the Distribution of the Suffixes by the Phonotactic Learner

The Phonotactic Learner learned a total of 227 constraints. Among them, three constraints are from the liquid tier: *[+rhotic][+rhotic]# (3.412), *[+lateral][+lateral] (1.079), *#[+lateral] (0.738) (the numbers are constraint weights). The first two constraints capture liquid dissimilation, as expected. The first constraint has a higher weight than the second constraint, indicating that avoiding two /r/s in an adjective is more important than avoiding two /l/s. This result conforms to the distribution patterns shown in Table 1. The suffix is never *-ar* if a stem contains /r/ non-finally, unless the stem final is /l/. Conversely, if a stem contains /l/ non-finally, the suffix can be *-al*.

As described, the 590 test words were 295 pairs of attested and unattested forms. The probability differences between attested and unattested forms are illustrated in Figure 4. The mean log-probability of attested forms is significantly higher (M=-2.79, SD=0.72) than the mean log-probability of unattested forms (M=-4.25, SD=1.01) ($t(534.04)=20.29$, $p<0.0001$). The model failed to predict correct labels for 31 pairs: higher probabilities were given to unattested forms than to attested forms in 5 pairs (1.7%), and equal probabilities were given to attested and unattested forms in 26 pairs (8.8%). Assuming the pairs with equal probabilities are incorrect, classification accuracy is 89.4% ((295-31)/295).

[6] The pronunciation form automatically generated by the CMU Pronouncing Dictionary. http://www.speech.cs.cmu.edu/cgibin/cmudict?in=palmer

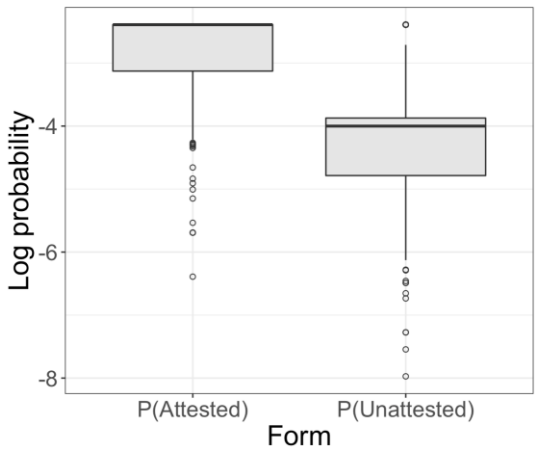


Figure 4. Log probabilities for attested and unattested forms computed by the PL

Table 2. Words and their log-probabilities where the PL failed to make correct predictions (N=5)

Suffix	Stem-final	Stem-internal	Log P (attested)	Log P (unattested)	Difference	Predicted suffix	Example	Number of words
al	Vowel	/l/	-3.1	-2.71	-0.47	*ar*	*ileal*	2
			-2.85	-2.39	-0.47		*intraepithelial*	
ar	Non-liquid C	No liquid	-4.21	-2.39	-1.82	*al*	*palmar*	1
		/l/	-4.29	-2.71	-1.85		*lumbar*	2
			-3.97	-2.39	-1.85		*prelumbar*	

Table 3. Words and their log-probabilities where the PL failed to determine the suffix (N=26)

Suffix	Stem-final	Stem-internal	Log P (attested)	Log P (unattested)	Difference	Example	Number of words	Total
al	Vowel	/l/	-2.71	-2.71	0	*lineal*	1	5
		/r/	-2.39	-2.39	0	*corneal*	3	
			-3.41	-3.41	0	*tracheal*	1	
	Non-liquid	/l/	-2.71	-2.71	0	*ileoanal*	1	15
			-2.39	-2.39	0	*polyclonal*	2	
		/r/	-2.39	-2.39	0	*marginal*	9	
		No liquid	-2.39	-2.39	0	*faunal*	3	
			-5.69	-5.69	0	*monumental*	1	
ar	Vowel	/l/	-2.71	-2.71	0	*linear*	1	2
			-2.39	-2.39	0	*rectilinear*	1	
	Non-liquid	/l/	-4.83	-4.83	0	*scapholunar*	1	3
			-2.71	-2.71	0	*laminar*	1	
			-2.39	-2.39	0	*triluminar*	1	

Table 2 shows the five words where the model gave higher probabilities to unattested forms. Table 3 shows the 26 words where attested and unattested forms were given equal probabilities. In Table 3, the model had the most errors for the words with non-liquid stem-final segments (15 words) while making no errors for those with liquid stem-final segments. The results above show that the PL model learned the invariable lateral dissimilation pattern occurring in the final syllable of the adjectives (*/l-əl#/, */r-ər#/). However, in non-liquid-final stems, the model failed to choose the correct suffixes in about 10.5% of the test words (31/295). In particular, the suffixes for 8.8%

of the test words were undetermined. This conforms to the distributional patterns observed in Section 3.1. The model's predictions are accurate where there are no variations and less accurate where the suffix varies, the condition where the stem-final segment is not liquid. The words with a non-liquid stem-final segment and stem-internal /r/ had the most errors (9 words).

3.3 Learning the Distribution of the Suffixes Using Deep Neural Networks

3.3.1 Classification accuracy

The ensemble model correctly predicted the suffixes for 287 stems out of the 295 words in the test set (test accuracy: (295-8)/295 = 97.3%). The model made wrong predictions for eight words, listed in Table 4. As in the PL, the deep learning model also had the most errors under the conditions where most variations are found. There are four error words when the stem final is a non-liquid consonant and /l/ is in the stem. For example, the suffix for *global* is predicted to be *-ar*.

Table 4. Error words in the deep learning model results(N=8)

Suffix	Stem-final	Stem-internal	Correct label	Predicted label	Mean Probability	Words
al	Vowel	/l/	*al*	*ar*	0.04	*intraepithelial* (1)
	Non-liquid C	/l/			0.29	*global, ileoanal, polyclonal, intraluminal* (4)
ar	Vowel	/l/	*ar*	*al*	0.56	*linear* (1)
	Non-liquid C	/l/			0.57	*laminar* (1)
		No liquid			0.95	*palmar* (1)

In the output of the DNN, stems with a probability higher than 0.5 are classified as *-al*-taking stems, and those with a probability less than 0.5 are classified as *-ar*-taking stems. The more different the probability from 0.5, the more confident the model is with the predicted label. If the probability is close to 0.5, the model is not highly confident of the predicted label. For example, the model assigned the suffix *-al* to *linear* and *laminar*, but it was not very confident with the label, considering the probabilities nearing 0.5 (0.56, 0.57, respectively). The vowel-final and /l/-internal stem is where most variations are found.

On the other hand, the model was highly confident with the incorrect labels for *intraepithelial* (0.04) and *palmar* (0.95). The error words from the DNN model are all included in the error words from the PL, except *global*. The PL assigned a higher P(al) than P(ar) for the word *global* (-3.3, -3.7).

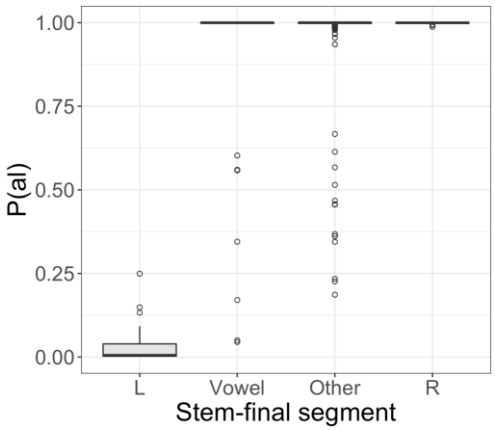


Figure 5. Predicted probability of *-al* depending on stem-final segments (/l/, vowel, non-liquid consonant, /r/). If P(al) < 0.5, the predicted suffix is *-ar*; otherwise, it is *-al*.

Figure 5 shows the predicted probability of *-al* depending on the stem-final segments. P(al) is the lowest when the stem-final segment is /l/, increasing in the order of vowel, non-liquid consonant, and /r/, which shows that the deep-learning model learned the pattern of liquid dissimilation as observed in the data. Among the outliers in the vowel-final stems, four words have an /l/ in the penultimate syllable (*intraepithelial* (0.045), *multinuclear* (0.050), *conciliar* (0.345), *ileal* (0.560)). For these words, P(al) is much lower than other vowel-final stems, suggesting the application of liquid dissimilation: when there is an /l/ in the penultimate syllable, P(al) is substantially low. When /r/ is the onset to the penultimate syllable, P(al) is 0.999 to 1 (e.g. *atrial*, *mercurial* (8 words)), suggesting the application of liquid dissimilation almost without exception. These predictions conform to the distributional patterns described in Section 3.1.

Figure 6 shows the number of suffixes predicted by the DNN for each condition. The predicted suffixes by the DNN are similar to the observed distributional patterns in Section 3.1, Figure 3. The proportion of the suffixes depending on the stem-final and stem-internal segments is almost identical to the attested distribution in Figure 3. That is, for the stem-final-/l/ condition, all suffixes are *-ar*; for the stem-final-/r/ condition, all suffixes are *-al*; the suffixes vary if the stem-final segment is a vowel or non-liquid consonant. A difference is that the exceptional words (*palmar*, *thenar*, *hyperthenar*) in the non-liquid stem-final ('Other') and no internal-liquid ('No liquid') conditions that existed in the attested data are not learned. *Palmar* is predicted to have the suffix *-al* (Table 4).

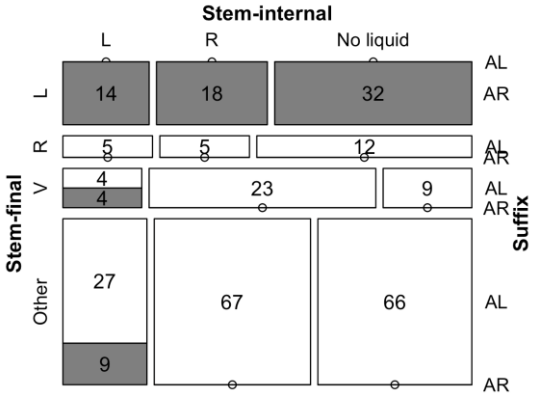


Figure 6. The number of words with *-al*/*-ar* suffixes as predicted by the DNN depending on their stem-final or internal liquids (white cells: *-al*, grey cells: *-ar*) (N=295). Small circles indicate zero frequency.

Table 5 shows the linear regression results with P(al) as a dependent variable and final and internal segments as predictors. The reference level was /l/-stem-final and /l/-stem-internal. The results show that stem-final segments that are not /l/ all significantly increase the probability of suffix *-al*, in the order of vowel, non-liquid consonant, and /r/, which exactly follows the distributional trend observed in Section 3.1.

Table 5. Linear regression results (P(al) ~ final × internal segments) (NLC means non-liquid consonants)

	B	S.E.	t	Pr(>|t|)	
(Intercept)	0.01	0.03	0.19	0.85	
final: /r/	0.99	0.06	16.99	<0.0001	***
final: Vowel	0.36	0.05	7.17	<0.0001	***
final: Non-liquid C	0.78	0.04	22.19	<0.0001	***
internal: /r/	0.01	0.04	0.36	0.72	
internal: Non-liquid C	0.03	0.04	0.85	0.40	
final /r/ × internal /r/	-0.01	0.08	-0.14	0.89	
final Vowel × internal /r/	0.62	0.06	10.26	<0.0001	***
final NLC × internal /r/	0.20	0.05	4.26	<0.0001	***
final /r/ × internal NLC	-0.03	0.07	-0.41	0.68	
final Vowel × internal NLC	0.61	0.07	9.33	<0.0001	***
final NLC × internal NLC	0.18	0.04	4.18	<0.0001	***

The coefficient for final /r/ (0.99) indicates the exceptionless application of liquid dissimilation when the final segment is a liquid. When the stem-final segment is /r/, P(al) is estimated to be 0.99. The coefficients for final Vowel and final NLC are all significantly different from zero. The coefficient is higher when the final segment is a non-liquid consonant, which reflects the distributional patterns in Figure 3. The proportion of suffix *-al* is higher when the final segment is a non-liquid than when it is a vowel.

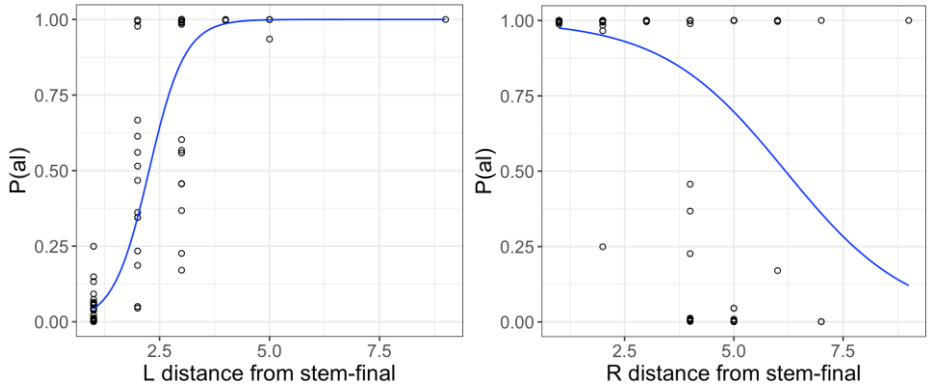
The interaction of final /r/ and internal /r/ seems to decrease P(al) (the negative coefficient, -0.01), contrary to the observations, but it is not significantly different from zero (p=0.89). In addition, the sum of the coefficients for final /r/ (0.99) and internal /r/ (0.01) is already a probability of 1.00, so the interaction of the two is not meaningful. That is, if a word has both final and internal /r/s, the P(al) of the word is 1.00. The same applies to the interaction of final /r/ and internal NLC (p=0.68).

The interaction between the final vowel and internal /r/ and the interaction between final NLC and internal /r/ are significantly different from zero (p<0.0001). The coefficient of the former is greater because the distance of /r/ to the suffix is always closer to the suffix in the former than in the latter. The stem-final vowel can take an /r/ onset in the penultimate syllable (e.g. *mercuri-al*). With a stem-final NLC, the /r/ is always further than this by at least one consonant (e.g. *marit-al*). Possibility of having /r/ closer to the vowel increases P(al), so the coefficient of the former (0.62) is greater than that of the latter (0.20). The same explanation can apply to the coefficient difference between the interaction of the final vowel and internal NLC and the interaction of final NLC and internal NLC.

Overall, the predicted probabilities of the deep-learning language model conform to what we observed in Section 3.1. The deep-learning model learned liquid dissimilation patterns in detail, with higher accuracy than the baseline model.

3.3.2 Distance effect

This section examines whether the DNN learned the distance effect on liquid dissimilation. The likelihood of applying liquid dissimilation decreases exponentially as the number of syllables increases between the liquid and the suffix (Zymet 2015). In our analysis, the distance was measured by the number of consonants counted leftward from the stem-final position (n^{th} liquid from the stem final segment), setting the stem-final position as 1. The stem-final liquid is distance 1 (e.g. *velar* [vil-ər]), and the one before the stem-final consonant is distance 2 (e.g. *algal* [æ lg-əl]). The distance of /l/ in *analogical* is 3 ([æ nəlɑʤik-əl], being the third consonant from the stem final segment. Vowels were counted only if they were stem-final (e.g. /r/ in *aerial* [ɛri̱-əl] is distance 2) to distinguish the stem-internal liquid from the stem-final liquid. Only the first occurrence of a liquid counted leftward from the stem-final position was considered (*lunular* [lunjəl-ər], distance 1). The distance was separately measured for /l/ and /r/ (named '/l/-distance' and '/r/-distance', respectively). For example, *translational* [træ nsleɪʃən-əl] is /l/-distance 3 and /r/-distance 6.



(a) (b)

Figure 7. Scatter plots showing the relations between P(al) and L-distance (a) and between P(al) and R-distance (b). The blue line is the fitted sigmoid curve.

Figure 7 shows the scatter plots showing the relations between P(al) and /l/-distance and between P(al) and /r/-distance. In Figure 7(a), P(al) increases as /l/-distance increases. The suffix *-al* is less likely if /l/ is closer to the stem-final position. In Figure 7(b), P(al) decreases as /r/-distance increases. The suffix *-al* is less likely if /r/ is farther from the stem-final position. Both plots thus conform to liquid dissimilation, but the slope of the sigmoid is steeper in Figure 7(a) than in Figure 7(b). This suggests that /l/-distance has more direct and gradient effects on P(al) than /r/-distance. In addition, the sigmoid curve is shifted leftward in Figure 7(a), suggesting that liquid dissimilation is rapidly and exponentially affected by the presence of an /l/ close to the stem-final position, as Zymet (2015) noted. When /l/ is stem-final (distance=1), P(al) is less than 0.25, so the suffix is never *-al*. However, when /l/ is in the syllable just preceding the suffix (distance=2), P(al) of 1.0 is already found. This conforms to the distribution patterns in Figure 3. The suffix varies when the stem final is non-liquid and the stem internal is /l/. Figure 7(b) also reflects the patterns observed in Figure 3. The datapoints in Figure 7(b) has either P(al)≈1.0 or

P(al)<0.5. If the stem internal is /r/ and the stem final segment is not /l/, the suffix is always *-al* (P(al)≈1.0). If the stem internal is /r/ and the stem final is /l/, the suffix is always *-ar* (P(al)<0.5), regardless of /r/-distance in Figure 7(b).

In terms of goodness of fit, McFadden's pseudo R^2 (ρ^2) (McFadden 1974) is 0.84 (Figure 7a) and 0.68 (Figure 7b). ρ^2 values in the range of 0.2-0.4 and higher suggest a very good fit (McFadden 1974, Louviere et al. 2000). Although both ρ^2 values are high, the fit for /l/-distance is better than that for /r/-distance. This once again suggests that the presence of /l/ in the stem is a more influential factor in the choice of the suffix compared to the presence of /r/.

3.3.3 Segmental blocking in terms of distances

The /l/ and /r/-distance measures can be used to analyze segmental blocking in dissimilation (Stanton 2017). Segmental blocking occurs when an /r/ intervenes between two /l/s (*plural*) or an /l/ intervenes between two /r/s (*regular*). In other words, regardless of the stem-internal liquids, the suffix is determined by the liquid closer to the suffix. The subtraction between /r/-distance and /l/-distance (/r/-distance - /l/-distance) can indicate which liquid is closer to the suffix and how much closer it is. If the difference is negative, /r/-distance is smaller than /l/-distance (/r/-distance < /l/-distance), i.e. /r/ is closer to the stem final position than /l/. For example, in *plural* [plʊr-əl], /r/-distance is 1, and /l/-distance is 2, resulting in a difference of -1. If the difference is positive, it means that /r/-distance is greater than /l/-distance, i.e., /l/ is closer to the stem final position than /r/. For example, in *regular* [rɛgjəl-ər], /r/-distance is 4, /l/-distance is 1, resulting in a difference of 3. Thus, we can expect that if the /r/-/l/ difference is negative, *-al* is more likely (P(al) increases), and if positive, *-al* is less likely (P(al) decreases). An inverse relation between P(al) and the /r/-/l/ difference is expected. Figure 8 shows the inverse relationship, as expected. The figure is drawn with only 40 words where /l/ and /r/ both exist in the stem.

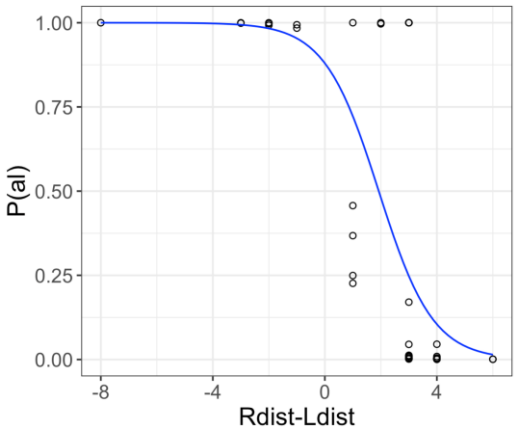


Figure 8. Scatter plot showing the relations between P(al) and the difference between R-distance and L-distance (N=40). The blue line is the fitted sigmoid curve. The distance difference can be obtained only in the words where both /l/ and /r/ exist in the stem.

In Figure 8, as the difference between /r/-distance and /l/-distance increases, P(al) decreases. When the difference is negative (i.e. /r/ is closer to the stem final than /l/), P(al) is 1 or close to 1. This suggests a segmental blocking effect where stem-final /r/ blocks the application of liquid dissimilation. When the difference is positive (i.e. /l/ is closer to the stem final position), P(al) is lower, reaching almost zero. McFadden's pseudo R^2 of the

fitted sigmoid curve in Figure 8 is 0.90.

To summarize, the relations between the probability of suffix *-al* and liquid distance show that the DNN learned the pattern of liquid dissimilation. The probability of *-al* rapidly decreases as /l/ draws close to the stem-final position. The probability of *-al* increases as /r/ draws close to the stem-final position. The probability of *-al* decreases when /r/ is closer to the stem final than /l/. These results indicate that segmental blocking is a gradient tendency. According to Hayes (2022), phonological variation falls into natural mathematical patterns and repeatedly follow a sigmoid curve. Our results also conform to this pattern. The probabilities learned by the DNN capture the natural mathematical pattern that exists in the choice of the suffix *-al* or *-ar*.

4. Discussion

This paper examines the variable distribution patterns of *-al* and *-ar* suffixes and whether the DNN can learn the distribution patterns. An examination of the adjectives reveals that when the stem-final segment is a liquid, the suffix is categorically determined. The suffix is *-ar* if the final segment is /l/, and the suffix is *-al* if the final segment is /r/. However, the suffix varies if the final segment is a vowel or a non-liquid consonant. The suffix is also affected by the stem-internal liquids. If there is no stem-final or internal /l/ but an internal /r/, the suffix is always *-al*. If the stem final is a non-liquid consonant with no liquid in the stem, the suffix is *-al*, with only a few exceptions.

The present study examined whether deep neural networks could learn these complicated patterns of suffix variation. We considered the morpho-phonemic variation as a case of a classification problem that is common in the field of machine learning. The deep neural networks first learned the patterns of the stems and their associated suffixes from the training data, and then the model classified the test data into *-al*-taking or *-ar*-taking stems. In the case of the PL, the probability of a stem to take *-ar* or *-al* was separately learned by using suffix-reversed forms, and the probabilities of attested/unattested forms were compared to determine the model's predicted suffixes.

In the results, the classification accuracy was higher in the DNN than in the PL (97.3% vs. 89.4%). The number of error words was higher in the PL than in the DNN (31 vs. 8 words). The two models correctly predicted the suffixes under the conditions where suffixes do not vary, i.e. those that have a liquid in the stem-final position. The two models made incorrect predictions where there were more variations in the data, i.e. non-liquid stem-final stems with stem-internal /l/s, but the number of error words was much smaller in the DNN results. Overall, the DNN was able to learn the distribution patterns more accurately than the PL.

In addition to the greater accuracy, the DNN enables a more precise quantitative confirmation of existing phenomena. The distance effect ('distance-based decay', Zymet (2015)) and the segmental blocking effect (Bennet 2015, Stanton 2017) were captured by the DNN model. The probability of taking the suffix *-al* gradually increases as /l/ appears farther from the stem-final position. The probability of *-al* gradually decreases as /r/ appears farther from the stem-final position. Yet, the DNN's predicted P(al) showed that /l/-distance has more direct effect on the suffix choice than /r/-distance, which has not been reported in the previous literature. The indicator of the segmental blocking, the /r/-/l/ distance, also showed a gradual decrease as the /r/-/l/ distance decreased, i.e. /r/ is farther from the stem-final position. If /r/ is farther from the stem-final position, it cannot block liquid dissimilation, and *-al* is less likely to attach. The probability values obtained from the DNN enable us to capture these distributional patterns in quantitatively precise ways.

Moreover, the patterns found by the DNN were fitted to the sigmoid curve, which is known to be frequent in

many linguistic phenomena, such as perception of voicing (Hayes 2022). The present result showed that the pattern of suffix distribution follows the sigmoid curve as in many other linguistic phenomena. At the same time, the result suggests that the DNN can learn linguistic patterns fitted in the sigmoid curve, so it can be effectively used to analyze linguistic data.

From the results so far, it is clear that the DNN can learn the suffix distributional patterns more accurately than the PL. The DNN utilized in this paper is one of the simplest and most basic ones among neural models, and other more advanced DNN models such as BERT (Bidirectional Encoder Representations from Transformers, Devlin et al. 2019) may achieve even greater accuracy in emulating human phonological knowledge. However, neural network models lack OT-style constraints that are interpretable by human linguists, which hinders their widespread use in phonological studies. In the present study, the DNN learns the distribution patterns directly from the data, while the PL learns the patterns through constraints. The present study showed how much we can learn from the data only, without the constraints. The DNN model, without the constraints, was able to learn more accurately than the model with the constraints. This result conforms to the previous research where a neural-network model without distinctive features outperformed the one with them (Mirea and Bicknell 2019). In contrast, the models with featural representations (e.g. the PL) performed better than the ones without them (e.g. bigram models) in modeling sonority projection (Daland et al. 2011). However, Daland et al. (2011) did not test neural-network models, and Mayer and Nelson (2020) showed that an RNN model performed better than the PL in learning sonority projection. Given this, it is worth reconsidering the roles of OT-style constraints and phonological features and the nature of these theoretical concepts. While contemplating this issue, we can continue to investigate the extent to which neural networks can approximate human knowledge. This, in itself, constitutes a valuable avenue of research and could serve as a potential direction of future studies.

5. Conclusion

As Petersen and Potts (2023) put it ("LLMs are powerful devices for studying lexical semantics in ways that can deeply inform linguistic theory"), deep neural networks can be powerful methods for studying phonotactic and phonological variable patterns. The present study exemplified how the DNN can be used to analyze the distribution of English suffixes with a complicated and probabilistic pattern. The DNN successfully learned the pattern of variation with high accuracy. Therefore, the DNN can be utilized to analyze phonotactic or phonological variable patterns, providing linguists with comprehensive insights that were previously unattainable through traditional theories.

References

Albright, A. 2009. Feature-based generalization as a source of gradient acceptability. *Phonology* 26, 9-41.

Beguš, G. 2020. Generative adversarial phonology: modeling unsupervised phonetic and phonological learning with neural networks. *Frontiers in Artificial Intelligence* 3, Article 44.

Bennett, W. G. 2015. *The Phonology of Consonants: Harmony, Dissimilation, and Correspondence*, Cambridge: Cambridge University Press.

Breiman, L. 2001. Random forests. *Machine Learning* 45, 5-32.

Carstairs-McCarthy, A. 2018. *An Introduction to English Morphology*. Edinburgh: Edinburgh University Press.

Cho, H. 2021. Predicting the gender of Korean personal names using fastText. *Studies in Phonetics, Phonology*

and Morphology 27(3), 483-500.
Chomsky, N. and M. Halle. 1968. *The Sound Pattern of English*. Cambridge: The MIT Press.
Cortes, C. and V. Vapnik. 1995. Support-vector networks. *Machine Learning* 20, 273-297.
Daland, R., B. Hayes, J. White, M. Garellek, A. Davis, I. and Normann. 2011. Explaining sonority projection effects. *Phonology* 28, 197-234.
Della Pietra, S., V. J. Della Pietra and J. D. Lafferty. 1997. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19, 380-393.
Devlin, J., M.-W. Chang, K. Lee and K. Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Volume 1 (Long and Short Papers), 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
Domingos, P.M. 1999. MetaCost: a general method for making classifiers cost-sensitive. In *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 155-164, San Diego, California.
Dressler, W. 1971. An alleged case of non-chronological rule insertion. *Linguistic Inquiry* 2, 597-599.
Fisher, R.A. 1936. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics* 7, 179-188.
Goldsmith, J. 1979. *Autosegmental Phonology*. New York: Garland.
Goldwater, S. and M. Johnson. 2003. Learning OT constraint rankings using a maximum entropy model. In *Proceedings of the Stockholm Workshop on Variation within Optimality Theory*, ed. by J. Spenader, A. Eriksson, and O. Dahl, 111–120. Stockholm: Stockholm University, Department of Linguistics.
Goodfellow, I., Y. Bengio and A. Courville. 2016. *Deep Learning*. The MIT Press.
Hayes, B. 2022. Deriving the Wug-shaped curve: A criterion for assessing formal theories of linguistic variation. *Annual Review of Linguistics* 8, 473-94.
Hayes, B. and C. Wilson. 2008. A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry* 39, 379-440.
Haykin, S. 1994. *Neural Networks: A Comprehensive Foundation*. 2nd ed. Delhi: Pearson Education.
Hochreiter, S. and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9(8), 1735-1780.
Howard, J. and S. Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Volume 1 (Long Papers), 328-339, Melbourne, Australia. Association for Computational Linguistics.
Kenstowicz, M. 1994. *Phonology in Generative Grammar*. Oxford: Blackwell.
Kim, K. H. 2021. Simple Neural Text Classification. Available online at https://github.com/kh-kim/simple-ntc
Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1746–1751, Doha, Qatar. Association for Computational Linguistics.
Krizhevsky, A., I. Sutskever and G. E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, Volume 1, 1097-1105. Lake Tahoe, Nevada.
LeCun, Y., L. Bottou, Y. Bensio and P. Haffner. 1998. Gradient based learning applied to document recognition. *Pro. IEEE* 305.
Linzen, T. 2019. What can linguistics and deep learning contribute to each other? Response to Pater. *Language* 95(1), e99-e108.
Louviere, J., D. Hensher, J. Swait and W. Adamowicz. 2000. *Stated Choice Methods: Analysis and Applications.* Cambridge: Cambridge University Press.
Mahowald, K. 2023. A discerning several thousand judgments: GPT-3 rates the article + adjective + numeral + noun construction. arXiv:2301.12564v2 [cs.CL]

Mayer, C. and M. Nelson. 2020. Phonotactic learning with neural language models. In *Proceedings of Society for Computation in Linguistics*. Volume 3, Issue 1, 149-159. https://aclanthology.org/2020.scil-1.36.pdf

McCallum, A. and K. Nigam. 1998. A Comparison of Event Models for Naive Bayes Text Classification. *Proceedings in Workshop on Learning for Text Categorization*, AAAI'98, 41-48.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. In P. Zarembka, ed., *Frontiers in Econometrics*, 105-142. New York: Academic Press.

Mikolov, T., K. Chen, G. Corrado and J. Dean. 2013. Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781[cs.CL]

Mirea, N. and K. Bicknell. 2019. Using LSTMs to assess the obligatoriness of phonological distinctive features for phonotactic learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1595-1605, Florence, Italy. Association for Computational Linguistics.

Schuster, M. and K. K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45(11), 2673-2681.

Park, K., S. You and S. Song. 2021. Not yet as native as native speakers: comparing deep learning predictions and human judgments. *English Language and Linguistics* 26(1), 199-228.

Pater, J. 2019. Generative linguistics and neural networks at 60: Foundation, friction, and fusion. *Language* 95(1), e41-e74.

Potts, C., J. Pater, K. Jesney, R. Bhatt and M. Becker. 2010. Harmonic Grammar with linear programming: From linear systems to linguistic typology. *Phonology* 27(1), 1-41.

Petersen, E. and C. Potts. 2023. Lexical Semantics with Large Language Models: A Case Study of English "break". In *Findings of the Association for Computational Linguistics: EACL 2023*, 490–511, Dubrovnik, Croatia. Association for Computational Linguistics.

Stanton, J. 2017. Segmental blocking in dissimilation: an argument for co-occurrence constraints. In *Proceedings of the Annual Meetings Phonology 2016*, Washington, DC.

Sundermeyer, M., H. Ney and R. Schlüter. 2015. From feedforward to recurrent LSTM neural networks for language modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3), 517-529.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser and I. Polosukhin. 2017. Attention is all you need. In *Proceedings of 31st Conference on Neural Information Processing Systems (NIPS 2017)*, 5998-6008, Long Beach, CA.

Wilcox, E. G., R. Futrell and R. Levy. 2023. Using computational models to test syntactic learnability. *Linguistic Inquiry* 2023, 1-88.

Zhang, L., S. Wang. and B. Liu. 2018. Deep Learning for Sentiment Analysis: A Survey. arXiv: 1801.07883 [cs.CL]

Zymet, J. 2015. Distance-Based Decay in Long-Distance Phonological Processes. In *Proceedings of the 32nd West Coast Conference on Formal Linguistics*,72-81, Somerville, MA.

Examples in: English
Applicable Languages: English
Applicable Level: Tertiary

