

Were We Already There? Applying Minimal Generalization to the SIGMORPHON-UniMorph Shared Task on Cognitively Plausible Morphological Inflection

Colin Wilson

Johns Hopkins University
colin.wilson@jhu.edu

Second Author

Affiliation / Address line 1
Affiliation / Address line 2
Affiliation / Address line 3
email@domain

Abstract

xxx

1 Introduction

In a landmark paper, Albright and Hayes (2003) proposed a model that learns morphological rules by recursive **minimal generalization** from lexeme-specific examples (e.g., $\text{ɪ} \rightarrow \text{ʌ} / \text{st} _ \eta$ for *sting* \sim *stung* and $\text{ɪ} \rightarrow \text{ʌ} / \text{fl} _ \eta$ for *fling* \sim *flung* generalized to $\text{ɪ} \rightarrow \text{ʌ} / \text{X} [-\text{syllabic}, +\text{coronal}, +\text{anterior}, \dots] _ \eta$).¹ The model was presented more formally in Albright and Hayes (2002), along with evidence that the rules it learns for the English past tense give a good account of native speakers' productions and ratings in wug-test experiments (e.g. judgments that *splung* is quite acceptable as the past tense of the novel verb *spling*). In addition to providing further analysis of the experimental data, Albright and Hayes (2003) compared their proposal with early connectionist models of morphology (e.g., Plunkett and Juola, 1999) and an analogical or 'family resemblance' model inspired by research on psychological categories (Nakisa et al., 2001).

Along with Albright (2002), which presents a parallel treatment of Italian inflection, Albright & Hayes's study of the English past tense is a paradigm example of theory-driven, multiple-methodology, open and reproducible research in cognitive science.² Their model has enduring significance for the study of morphological learning

and productivity in English (e.g., Rácz et al., 2014, 2020; Corkery et al., 2019) and many other languages (e.g., Hijazi Arabic: Ahyad 2019; Japanese: Oseki et al. 2019; Korean: Albright and Kang 2009; Navajo: Albright and Hayes 2006; Portuguese: Veríssimo and Clahsen 2014; Russian: Kapatsinski 2010; Tgdaya Seediq: Kuo 2020; Spanish: Albright and Hayes 2003; Swedish: Strik 2014).

In this paper, we apply a partial reimplementa-tion of the Albright & Hayes model to wug-test rating data from three languages (German, English, and Dutch) collected for the SIGMORPHON-UniMorph 2021 Shared Task. Our version of the model is based purely on minimal generalization of morphological rules, as described in §3.1 of Albright and Hayes (2002) and reviewed below. It does not include additional mechanisms for learning phonological rules, and expanding or reigning in morphological rules, that were part of the original model (see Albright and Hayes, 2002, §3.3 - §3.7). We think there is much to be gained from considering minimal generalization on its own, with the other mechanisms ablated away, as borne out by competitive results on the shared task. For convenience, we refer to the original model as **min-gen** and our bare-bones version as **mingen0**.

1.1 Outline

xxx

2 Minimal Generalization

2.1 Inputs

The model takes as input a set of wordform pairs, one per lexeme, that instantiate the same morphological relationship. In simulations of English past tense formation, these are pairs of bare verb stems and past tense forms such as $\langle \text{ʌwɔk}, \text{ʌwɔkt} \rangle$, $\langle \text{ʌtɔk}, \text{ʌtɔkt} \rangle$, $\langle \text{ʌstɪŋ}, \text{ʌstɪŋ} \rangle$, $\langle \text{ʌflɪŋ}, \text{ʌflɪŋ} \rangle$, and $\langle \text{ʌkʌt}, \text{ʌkʌt} \rangle$. Wordforms consist of phonological segments (here, in broad transcription) delimited by special beginning and end

¹The square brackets contain the shared phonological feature specifications of /t/ and /l/, which in the feature system used here are xxx.

²Albright & Hayes released both the results of their wug-test experiments and an implementation of their model (visit <http://www.mit.edu/~albright/mgl/> and <https://linguistics.ucla.edu/people/hayes/RulesVsAnalogy/index.html>). An impediment to large-scale simulation with the model is that it runs from a GUI interface only. As part of the present project, we have added a command line interface to the original source code (available on request).

of string symbols. The set Σ of phonological segments, and the set $\Sigma_{\#} = \Sigma \cup \{\bowtie, \bowtie\}$, are assumed to be given to the model.

The model also requires a phonological feature specification for each of the symbols that appears in wordforms. We used a well-known feature chart, augmenting it with orthogonal feature specifications for the delimiters \bowtie and \bowtie .³ Φ is the set of all possible (partial) feature specifications over the chosen set and $\phi(x)$ gives the features of $x \in \Sigma_{\#}$.

2.2 Base rules

For each wordform pair, the model constructs a lexeme-specific morphological rule by first identifying the longest common prefix (lcp) of the wordforms excluding \bowtie (C), then the longest common suffix from the remainder (D), and finally identifying the remaining symbols in the first (A) and second (B) members of the pair. The symbol $\emptyset \notin \Sigma_{\#}$ denotes the empty string.⁴ The rule formed from $\langle \bowtie w\acute{o}k\bowtie, \bowtie w\acute{o}k\bowtie \rangle$ has the components $C = \bowtie w\acute{o}k$, $D = \bowtie$, $A = \emptyset$ and $B = t$ (i.e., $\emptyset \rightarrow t / \bowtie w\acute{o}k _ \bowtie$). The rule for $\langle \bowtie k\acute{a}t\bowtie, \bowtie k\acute{a}t\bowtie \rangle$ is $\emptyset \rightarrow \emptyset \bowtie k\acute{a}t _ \bowtie$.

2.3 Minimal Generalization

Given any two base rules R_1 and R_2 that make the same change ($A \rightarrow B$), the model forms a possibly more general rule by aligning and comparing their contexts. The minimal generalization operation, $R = R_1 \sqcap R_2$, carries over the common change of the two base rules and applies independently to their left-hand (C_1, C_2) and right-hand (D_1, D_2) contexts. For convenience, we define minimal generalization of the right-hand contexts. Minimal generalization of the left-hand contexts can be performed by reversing C_1 and C_2 , applying the definition for right-hand contexts, and reversing the result.

The minimal generalization $D = D_1 \sqcap D_2$ is defined procedurally by first extracting the lcp $\sigma_{1\wedge 2}$ of the two contexts and then operating on the remainders (D'_1, D'_2). If both D'_1 and D'_2 are empty then $D = \sigma_{1\wedge 2}$. If one but not both of them are

empty then $D = \sigma_{1\wedge 2}X$, where $X \notin \Sigma_{\#}$ is a variable over symbol sequences (i.e., X stands for $\Sigma_{\#}^*$). If neither is empty, then the operation determines whether their initial symbols have any shared features; for this purpose it is useful to consider $\phi(x)$ as a function from symbols to sets of feature-value pairs, in which case the common features are found by set intersection.

If there are no common features, $\phi_{1\cap 2} = \emptyset$, then as before $D = \sigma_{1\wedge 2}X$. Otherwise, the set of common features $\phi_{1\cap 2} \neq \emptyset$ is appended to $\sigma_{1\wedge 2}$, the first symbol is removed from D'_1 and D'_2 , and the operation applies to the remainders. If both remainders are empty then $D = \sigma_{1\wedge 2}\phi_{1\cap 2}$, otherwise $D = \sigma_{1\wedge 2}\phi_{1\cap 2}X$.

In summary, the generalized right-hand context D consists of the longest common prefix shared by D_1 and D_2 , followed by a single set of shared features (if any), followed by X in case there are no shared features or one context is longer than the other. With the change and generalized left-hand context C determined as already described, the result of applying minimal generalization to the two base rules is $R = A \rightarrow B/C _ D$.⁵

2.4 Recursive Minimal Generalization

Let \mathcal{R}_1 be the set of base rules (one per wordform pair in the input data) and \mathcal{R}_2 be the set containing all of the base rules and the result of applying minimal generalization to each eligible pair of base rules. While the rules of \mathcal{R}_2 have greater collective scope, they are nevertheless unlikely to account for the level of morphological productivity shown by native speakers. For example, English speakers can systematically rate and produce past tense forms of novel verbs that contain unusual segment sequences, such as *ploomf* /*ploomf*/ (e.g., [Prasada and Pinker, 1993](#)). Albright & Hayes propose to apply minimal generalization recursively and demonstrate that this can yield rules of great generality (e.g., in our notation, $\emptyset \rightarrow t / X [-voice] _ \bowtie$).

In the original proposal, recursive minimal generalization was defined only for pairs that include one base rule; it was conjectured that no additional

³The phonological feature chart is available from Bruce Hayes's website, <https://linguistics.ucla.edu/people/hayes/120a/Index.htm#features>. xxx binary with 0s xxx original features distributed with the model included scalar features xxx for example. Alternative binary (with underspecification) feature sets are xxx phoible ([Moran et al., 2014](#)) xxx panphon ([Mortensen et al., 2016](#)).

⁴In other common notations, the empty string is denoted by λ . xxx notation for phonological segment strings generally follows (?) and works cited there.

⁵There could be a slight difference between our definition of context generalization and that in [Albright and Hayes \(2002\)](#), hinging on whether the empty feature set is allowed in rules. In our definition, $\phi_{1\cap 2} = \emptyset$ is replaced by the variable X . It is possible that the original proposal intended for empty and non-empty feature sets to be treated alike. The definitions can diverge when applied to contexts that are of identical length and share all but the last (resp. first) segments, in which case our version would result in a broader rule.

generalizations could result from dropping this restriction. Here we define the operation for any two right-hand contexts $D_1, D_2 \in \Sigma_{\#}^*(\Phi)(X)$. As before, only rules that make the same change are eligible for generalization and the operation applies to left-hand contexts under reversal.

The revised definition of $D = D_1 \sqcap D_2$ is identical to that given above except that we must consider input contexts that contain feature sets and X (which previously could occur only in outputs). As before, we first identify the lcp of symbols from $\Sigma_{\#}$ in the two contexts, $\sigma_{1\wedge 2}$, and then operate on the remainders (D'_1, D'_2). If both D'_1 and D'_2 are empty then $D = \sigma_{1\wedge 2}$. If one but not both of them are empty then $D = \sigma_{1\wedge 2}X$. If both are non-empty then their initial elements are either symbols in $\Sigma_{\#}$, feature sets in Φ , or X . Replace any initial symbol $x \in \Sigma_{\#}$ with its feature set $\phi(x)$, extend the function ϕ so that $\phi(X) = \emptyset$, and compute the unification $\phi_{1\cap 2}$ of the initial elements. The rest of the definition is unchanged (see the end of §2.3).

By construction, the contexts that result from this operation are also in $\Sigma_{\#}^*(\Phi)(X)$ (i.e., no ordinary symbol can occur after a feature set, there is at most one feature set, X can occur only at the end of the context, etc.). Therefore, the revised definition supports the application of minimal generalization to its own products. Let \mathcal{R}_k be the set of rules containing every member of \mathcal{R}_{k-1} and the result of applying minimal generalization to each eligible pair of rules in \mathcal{R}_{k-1} (for $k > 1$). In principle, there is an infinite sequence of rules set related by inclusion $\mathcal{R}_1 \subseteq \mathcal{R}_2 \subseteq \mathcal{R}_3 \dots$. In practice, the equality becomes strict after a small number of iterations of minimal generalization (typically 6-7), at which point there are no more rules to be found.

2.5 Completeness

Having defined minimal generalization for arbitrary contexts (as allowed by the model), we can revisit the conjecture that nothing is lost by restricting the operation to pairs at least one of which is a base rule. This is a practical concern, as the number of base rules is a constant determined by the input data while the number of generalized rules can increase exponentially.

Conceptually, each rule learned by unrestricted minimal generalization has a (possibly non-unique) ‘history’ of base rules from which it originated. A base rule $R \in \mathcal{R}_1$ has the history $\{R\}$. A rule in $R \in \mathcal{R}_2$ has the history $\{R_1, R_2\}$ containing the

two base rules from which it derived. In general, the history of each rule in \mathcal{R}_k is the union of the histories of two rules in \mathcal{R}_{k-1} ($k > 1$).

Because all rules are learned ‘bottom-up’ in this sense, the conjecture can be proved by showing that the minimal generalization operation is associative; we also show that it is commutative — both properties inherited from equality, lcp, set intersection, and other more primitive ingredients. As before, we explicitly consider right-hand contexts, from which parallel results for left-hand contexts and entire rules follow immediately. It follows that any rule R can be replaced, for the purposes of minimal generalization, with the base rules in its history (in any order).

Commutative. Let $D = D_1 \sqcap D_2$ for any $D_1, D_2 \in \Sigma_{\#}^*(\Phi)(X)$. We prove by construction that D is also equal to $D_2 \sqcap D_1$. The lcp of elements from $\Sigma_{\#}$ is the same regardless of the order of the contexts ($\sigma_{1\wedge 2} = \sigma_{2\wedge 1}$) as are the remainders (D'_1 and D'_2). If both remainders are empty, then the result of minimal generalization is $\sigma_{1\wedge 2} = \sigma_{2\wedge 1}$. If one but not both of them are empty then the result is $\sigma_{1\wedge 2}X = \sigma_{2\wedge 1}X$; note that X appears regardless of which context is longer. If both are non-empty then we ensure that their initial elements are (possibly empty) feature sets and take their intersection, which is order independent: $\phi_{1\cap 2} = \phi_{2\cap 1}$. If $\phi_{1\cap 2} = \emptyset$ then the result is $\sigma_{1\wedge 2}X = \sigma_{2\wedge 1}X$. Otherwise, the initial elements are removed and the operation continues to the remainders. If both remainders are empty the result is $\sigma_{1\wedge 2}\phi_{1\cap 2} = \sigma_{2\wedge 1}\phi_{2\cap 1}$, otherwise it is the same expressions terminated by X .

Associative. Let $D = (D_1 \sqcap D_2) \sqcap D_3$ for any $D_1, D_2, D_3 \in \Sigma_{\#}^*(\Phi)(X)$. We prove by construction that D is equal to $E = D_1 \sqcap (D_2 \sqcap D_3)$. Let σ be the longest prefix of symbols from $\Sigma_{\#}$ in D . Because σ occurs in D iff it is the lcp of this type in $(D_1 \sqcap D_2)$ and D_3 , it must be a prefix of each of D_1, D_2, D_3 and the longest such prefix in at least one of them. It follows that σ is also the lcp of symbols from $\Sigma_{\#}$ in D_1 and $(D_2 \sqcap D_3)$. Therefore, D and E both begin with σ . We now remove the prefix σ from all of the input contexts and consider the remainders D'_1, D'_2, D'_3 .

If all of the remainders are empty, then $D = E = \sigma$. If all but one of them are empty, then $D = E = \sigma X$.⁶ If none of the remainder is empty,

⁶If D'_1 or D'_2 is the longest context, assume by commutativity that it is D'_1 . The minimal generalizations are

let ϕ_1, ϕ_2, ϕ_3 be their (featurized) initial elements. The intersection of those elements is independent of grouping, $\phi = (\phi_1 \cap \phi_2) \cap \phi_3 = \phi_1 \cap (\phi_2 \cap \phi_3)$. If the intersection is empty then again $D = E = \sigma X$. If the intersection is non-empty then D and E both begin $\sigma\phi$. Finally, remove the initial elements of each of D'_1, D'_2, D'_3 and compare the lengths of the remainders to determine whether X appears at the end of D and E ; this is independent of grouping along the same lines shown previously.

Complete. We now prove by induction that, for any $R \in \mathcal{R}_k$ and $R_1, R_2 \in \mathcal{R}_{k-1}$ ($k > 1$) such that $R = R_1 \sqcap R_2$, rule R can also be derived by applying minimal generalization to R_1 and one or more base rules (*i.e.*, the rules in the history of R_2).⁷ For $R \in \mathcal{R}_2$ this is true by definition. For $R \in \mathcal{R}_3$, we have $R = R_1 \sqcap R_2 = R_1 \sqcap (R_{21} \sqcap R_{22}) = (R_1 \sqcap R_{21}) \sqcap R_{22}$, where R_{21} and R_{22} are base rules whose minimal generalization results is R_2 . In general, suppose that the statement is true for $k - 1 > 0$. Then it is also true for k because $R \in \mathcal{R}_k$ can be derived by $R_1 \sqcap R_2 = R_1 \sqcap (\cap_{i=1}^n R_{2i}) = (((R_1 \sqcap R_{21}) \sqcap R_{22}) \cdots \sqcap R_{2n})$ where $R_1, R_2 \in \mathcal{R}_{k-1}$ and each R_{2i} is a base rule in the history of R_2 .

These results validate the rule learning algorithm proposed by Albright and Hayes (2002) and used in our implementation. Any minimal generalization of two arbitrary rules R_1 and R_2 (as allowed by the model) can also be derived from R_1 (or R_2) by recursive application of minimal generalization with one or more base rules.

3 System Description

Our system for the shared task preprocessed the input wordforms, learned rules recursively as in §2, scored the rules in two ways, pruned out rule that have no effect on the model’s predictions, and applied the remaining rules to wug forms in order to generated predicted ratings.

3.1 Preprocessing

The shared task provided space-separated broad IPA transcriptions of the training and wug wordforms (*e.g.*, s t ɪ ŋ, s t ʌ ŋ, w ɔ k, w ɔ k t). As already mentioned, we added explicit beginning and end of string symbols. Because minimal generalization

⁷ $(D'_1 \sqcap D'_2) = X$ and $X \sqcap D'_3 = X$, which gives the same result as $(D'_2 \sqcap D'_3) = \lambda$ and $D'_1 \sqcap \lambda = X$. Similar reasoning applies if D'_3 is the longest context.

⁷We ignore rules that are carried over from \mathcal{R}_{k-1} to \mathcal{R}_k .

requires each wordform symbol to have a phonological feature specification, but some segments in the data lack entries in our feature chart, we further simplified or split the symbols as follows.

For German, we split the diphthongs /aɪ aʊ oɪ i:ə e:ə ɛ:ə/ into their component vowels and additionally regularized /i̯ u̯/ to /i u/. For English, we split the diphthongs /aɪ aʊ ɔɪ uɪ/ into their components and /ɜ:/ into /ɛ ɪ/, simplified /eɪ əʊ/ to /e o/, and regularized /m̩ n̩ r̩ l̩ ʃ̩/ to /m n ɪ l ɔ/. We also deleted all length marks /:/ and instances of /ʸ/. For Dutch, we split /ɛɪ aʊ ʊɪ/ into their components.

Checking that all wordform symbols appear in a phonological feature chart is also useful for data cleaning. It helped us to identify a few thousand Dutch wordforms containing ‘+’ (indicating a Verb - Preposition combination), which we removed. And it caught an encoding error in which two distinct but perceptually similar Unicode symbols were used for /g/.

Two acknowledged limitations of the original version of the minimal generalization model, and our version, are relevant here. First, the model learns rules for individual morphological relations (*e.g.*, mapping a bare stem to a past tense form), not for entire morphological systems jointly. Therefore, we retained from the preprocessed input data only the wordform pairs that instantiate the relations targeted by the wug tests: formation of past participles in German and past tenses in English and Dutch.

Second, the model cannot learn sensible rules for circumfixes (xxx). This could be remedied by allowing the model to form rules that simultaneously make changes at both edges of inputs, or by allowing it to apply multiple single-edge rules when mapping inputs to outputs. As a provisional solution, we removed the invariant prefix /gə-/ whenever it occurred at the beginning of a German past participle (training or wug wordform).

3.2 Scoring

xxx

3.3 Pruning

When applied to a data set containing thousands of lexemes, recursive minimal generalization can produce

3.4 Application

(Gorman, 2016; Gorman and Sproat, 2021) xxx

(Hayes and Wilson, 2008)

3.5 Prediction

4 Results

Clahsen (1999)

Booij (2019)

5 Conclusions and Future Directions

(Bybee and Moder, 1983)

(Tenenbaum, 1999)

(Plotkin, 1970)

These instructions are for authors submitting papers to *ACL conferences using L^AT_EX. They are not self-contained. All authors must follow the general instructions for *ACL proceedings,⁸ and this document contains additional instructions for the L^AT_EX style files.

The templates include the L^AT_EX source of this document (`acl.tex`), the L^AT_EX style file used to format it (`acl.sty`), an ACL bibliography style (`acl_natbib.bst`), an example bibliography (`custom.bib`), and the bibliography for the ACL Anthology (`anthology.bib`).

6 Engines

To produce a PDF file, pdfL^AT_EX is strongly recommended (over original L^AT_EX plus `dvips+ps2pdf` or `dvipdf`). XeL^AT_EX also produces PDF files, and is especially suitable for text in non-Latin scripts.

7 Preamble

The first line of the file must be

```
\documentclass[11pt]{article}
```

To load the style file in the review version:

```
\usepackage[review]{acl}
```

For the final version, omit the `review` option:

```
\usepackage{acl}
```

To use Times Roman, put the following in the preamble:

```
\usepackage{times}
```

(Alternatives like `txfonts` or `newtx` are also acceptable.)

Please see the L^AT_EX source of this document for comments on other packages that may be useful.

Set the title and author using `\title` and `\author`. Within the author list, format multiple authors using `\and` and `\And` and `\AND`; please see the L^AT_EX source for examples.

By default, the box containing the title and author names is set to the minimum of 5 cm. If you need more space, include the following in the preamble:

```
\setlength\titlebox{<dim>}
```

where `<dim>` is replaced with a length. Do not set this length smaller than 5 cm.

⁸<http://acl-org.github.io/ACLPUB/formatting.html>

Command	Output	Command	Output
<code>\a</code>	ä	<code>\c c</code>	ç
<code>\^e</code>	ê	<code>\u g</code>	ğ
<code>\i</code>	ì	<code>\l</code>	ł
<code>\.I</code>	İ	<code>\~n</code>	ñ
<code>\o</code>	ø	<code>\H o</code>	ő
<code>\'u</code>	ú	<code>\v r</code>	ř
<code>\aa</code>	å	<code>\ss</code>	ß

Table 1: Example commands for accented characters, to be used in, e.g., Bib_{TEX} entries.

8 Document Body

8.1 Footnotes

Footnotes are inserted with the `\footnote` command.⁹

8.2 Tables and figures

See Table 1 for an example of a table and its caption. **Do not override the default caption sizes.**

8.3 Hyperlinks

Users of older versions of L_AT_EX may encounter the following error during compilation:

```
\pdfendlink ended up in
different nesting level
than \pdfstartlink.
```

This happens when pdfL_AT_EX is used and a citation splits across a page boundary. The best way to fix this is to upgrade L_AT_EX to 2018-12-01 or later.

8.4 Citations

Table 2 shows the syntax supported by the style files. We encourage you to use the natbib styles. You can use the command `\citet` (cite in text) to get “author (year)” citations, like this citation to a paper by ?. You can use the command `\citep` (cite in parentheses) to get “(author, year)” citations (?). You can use the command `\citealp` (alternative cite without parentheses) to get “author, year” citations, which is useful for using citations within parentheses (e.g. ?).

8.5 References

The L_AT_EX and Bib_{TEX} style files provided roughly follow the American Psychological Association format. If your own bib file is named `custom.bib`, then placing the following before any appendices in

your L_AT_EX file will generate the references section for you:

```
\bibliographystyle{acl_natbib}
\bibliography{custom}
```

You can obtain the complete ACL Anthology as a Bib_{TEX} file from <https://aclweb.org/anthology/anthology.bib.gz>. To include both the Anthology and your own .bib file, use the following instead of the above.

```
\bibliographystyle{acl_natbib}
\bibliography{anthology, custom}
```

Please see Section 9 for information on preparing Bib_{TEX} files.

8.6 Appendices

Use `\appendix` before any appendix section to switch the section numbering over to letters. See Appendix A for an example.

9 Bib_{TEX} Files

Unicode cannot be used in Bib_{TEX} entries, and some ways of typing special characters can disrupt Bib_{TEX}’s alphabetization. The recommended way of typing special characters is shown in Table 1.

Please ensure that Bib_{TEX} records contain DOIs or URLs when possible, and for all the ACL materials that you reference. Use the `doi` field for DOIs and the `url` field for URLs. If a Bib_{TEX} entry has a URL or DOI field, the paper title in the references section will appear as a hyperlink to the paper, using the `hyperref` L_AT_EX package.

Acknowledgements

This research was partially supported by NSF grant BCS-1844780.

References

- Honaida Yousuf Ahyad. 2019. *Vowel Distribution in the Hijazi Arabic Root*. Ph.D. dissertation, State University of New York at Stony Brook.
- Adam Albright. 2002. Islands of reliability for regular morphology: Evidence from Italian. *Language*, 78(4):684–709.
- Adam Albright and Bruce Hayes. 2002. [Modeling English past tense intuitions with minimal generalization](#). In *Proceedings of the ACL-02 Workshop on Morphological and Phonological Learning*, pages 58–69. Association for Computational Linguistics.

⁹This is a footnote.

Table 2: Citation commands supported by the style file. The style is based on the natbib package and supports all natbib citation commands. It also supports commands defined in previous ACL style files for compatibility.

- Adam Albright and Bruce Hayes. 2003. [Rules vs. analogy in English past tenses: A computational/experimental study](#). *Cognition*, 90(2):119–161.
- Adam Albright and Bruce Hayes. 2006. Modeling productivity with the Gradual Learning Algorithm: The problem of accidentally exceptionless generalizations. In Gisbert Fanselow, Caroline Fery, Matthias Schlesewsky, and Ralf Vogel, editors, *Gradience in Grammar: Generative Perspectives*, pages 185–204. Oxford University Press, Oxford.
- Adam Albright and Yoonjung Kang. 2009. Predicting innovative alternations in Korean verb paradigms. *Current issues in unity and diversity of languages: Collection of the papers selected from the 18th International Conference of Linguistics*, pages 1–20.
- Geert Booij. 2019. *The Morphology of Dutch*, second edition. Oxford University Press, New York.
- Benjamin Börschinger and Mark Johnson. 2011. [A particle filter algorithm for Bayesian wordsegmentation](#). In *Proceedings of the Australasian Language Technology Association Workshop 2011*, pages 10–18, Canberra, Australia.
- Joan L. Bybee and Carol Lynn Moder. 1983. [Morphological classes as natural categories](#). *Language*, 59(2):251–270.
- Harald Clahsen. 1999. [Lexical entries and rules of language: A multidisciplinary study of German inflection](#). *Behavioral and Brain Sciences*, 22(6):991–1013.
- Maria Corkery, Yevgen Matuskevych, and Sharon Goldwater. 2019. [Are we there yet? encoder-decoder neural networks as cognitive models of English past tense inflection](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3868–3877, Florence, Italy. Association for Computational Linguistics.
- James Goodman, Andreas Vlachos, and Jason Naradowsky. 2016. [Noise reduction and targeted exploration in imitation learning for Abstract Meaning Representation parsing](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–11, Berlin, Germany. Association for Computational Linguistics.
- Kyle Gorman. 2016. [Pynini: A Python library for weighted finite-state grammar compilation](#). In *Proceedings of the SIGFSM Workshop on Statistical NLP and Weighted Automata*, pages 75–80, Berlin, Germany. Association for Computational Linguistics.
- Kyle Gorman and Richard Sproat. 2021. [Finite-State Text Processing](#). *Synthesis Lectures on Human Language Technologies*, 14(2):1–158.
- Mary Harper. 2014. [Learning from 26 languages: Program management and science in the babel program](#). In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, page 1, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.
- Bruce Hayes and Colin Wilson. 2008. [A maximum entropy model of phonotactics and phonotactic learning](#). *Linguistic Inquiry*, 39(3):379–440.
- Vsevolod Kapatsinski. 2010. [Velar palatalization in Russian and artificial grammar: Constraints on models of morphophonology](#). *Laboratory Phonology*, 1(2):361–393.
- Jennifer Kuo. 2020. *Evidence for Base-Driven Alternation in Tgdaya Seediq*. Master’s Thesis, UCLA.
- Steven Moran, Daniel McCloy, and Richard Wright. 2014. PHOIBLE Online. *PHOIBLE Online*. Edited by: Moran, Steven; McCloy, Daniel; Wright, Richard (2014). Leipzig: Max Planck Institute for Evolutionary Anthropology.
- David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. [PanPhon: A resource for mapping IPA segments to articulatory feature vectors](#). In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3475–3484, Osaka, Japan. The COLING 2016 Organizing Committee.
- Ramin Charles Nakisa, Kim Plunkett, and Ulrika Hahn. 2001. A cross-linguistic comparison of single and dual-route models of inflectional morphology. In Peter Broeder and Jaap Murre, editors, *Models of Language Acquisition: Inductive and Deductive Approaches*. MIT Press, Cambridge, MA.
- Yohei Oseki, Yasutada Sudo, Hiromu Sakai, and Alec Marantz. 2019. [Inverting and modeling morphological inflection](#). In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 170–177, Florence, Italy. Association for Computational Linguistics.
- Gordon D. Plotkin. 1970. A note on inductive generalization. In *Machine Intelligence*, volume 5, pages 153–163. Edinburgh University Press.
- Kim Plunkett and Patrick Juola. 1999. [A connectionist model of English past tense and plural morphology](#). *Cognitive Science*, 23(4):463–490.

- Sandeep Prasada and Steven Pinker. 1993. Generalisation of regular and irregular morphological patterns. *Language and Cognitive Processes*, 8(1):1–56.
- Péter Rácz, Clay Beckner, Jennifer B. Hay, and Janet B. Pierrehumbert. 2020. [Morphological convergence as on-line lexical analogy](#). *Language*, 96(4):735–770.
- Péter Rácz, Clayton Beckner, Jennifer B. Hay, and Janet B. Pierrehumbert. 2014. [Rules, analogy, and social factors codetermine past-tense formation patterns in English](#). In *Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM*, pages 55–63, Baltimore, Maryland. Association for Computational Linguistics.
- Oscar Strik. 2014. [Explaining tense marking changes in Swedish verbs: An application of two analogical computer models](#). *Journal of Historical Linguistics*, 4(2):192–231.
- Joshua Tenenbaum. 1999. [Bayesian modeling of human concept learning](#). In *Advances in Neural Information Processing Systems*, volume 11, Cambridge, MA. MIT Press.
- João Veríssimo and Harald Clahsen. 2014. [Variables and similarity in linguistic generalization: Evidence from inflectional classes in Portuguese](#). *Journal of Memory and Language*, 76:61–79.

A Example Appendix

This is an appendix.