Do transformer models do phonology like a linguist?

Saliha Muradoğlu[♪♩] **Mans Hulden**[♫]
[♪]The Australian National University (ANU) [♫]University of Colorado
[♩]ARC Centre of Excellence for the Dynamics of Language (CoEDL)
saliha.muradoglu@anu.edu.au, mans.hulden@colorado.edu

Abstract

Neural sequence-to-sequence models have been very successful at tasks in phonology and morphology that seemingly require a capacity for intricate linguistic generalisations. In this paper, we perform a detailed breakdown of the power of such models to capture various phonological generalisations and to benefit from exposure to one phonological rule to infer the behaviour of another similar rule. We present two types of experiments, one of which establishes the efficacy of the transformer model on 29 different processes. The second experiment type follows a priming and held-out case split where our model is exposed to two (or more) phenomena; one which is used as a primer to make the model aware of a linguistic category (e.g. voiceless stops) and a second one which contains a rule with a withheld case that the model is expected to infer (e.g. word-final devoicing with a missing training example such as **b**→**p**). Our results show that the transformer model can successfully model all 29 phonological phenomena considered, regardless of perceived process difficulty. We also show that the model can generalise linguistic categories and structures, such as vowels and syllables, through priming processes.

1 Introduction

In computational linguistics, neural networks have occupied much of recent work. One prime driver is adaptability to multiple facets of linguistic phenomena. As an example, sequence-to-sequence models have been shown to capture inflection patterns across numerous languages (Kodner et al., 2022). While their performance represents significant advances, the abstractions generated during the modelling process warrant further investigation. We experiment with phonological processes on a constructed language to compare the generalisations learned by transformer models with widespread linguistic phenomena.

In particular, we address the following questions:

- Learning specific phonological processes (are some more difficult than others?)
- Categorisation (can the model generalise a category, vowels, consonants, specific consonant groups, e.g. plosives?)
- Is word structure (syllables) implicitly learned?

We establish that the transformer model successfully models all 29 phonological phenomena we consider, regardless of linguistic complexity. Our results show that the model can generalise to linguistic categories with some caveats. By examining the transformer model's generalisation of haplology, we show that the model appears to learn syllables; the model can recognise the difference between VC and CV and generate previously unseen CV sequences.

2 Related Work

Investigating the cognitive reality of linguistic categories defined within phonology has long been of interest to linguistics. Does the natural class of phonemes bear any significance to a cognitive reality? For example, a series of experiments (Finley and Badecker, 2009; Chambers et al., 2010; Skoruppa and Peperkamp, 2011) examine the natural class of vowels and whether phonological patterns can be extended to previously unseen vowels. The studies suggest that participants were mostly able to generalise. In a similar vein, Finley (2011) presents a study on consonant harmony. The results suggest that learners (human learners) can generalise to novel consonants when the phonological pattern is general. However, the learners failed to generalise when the rule triggering the consonant harmony pattern was highly specific.

We adapt this long-standing linguistic question to ask whether Transformer-based abstractions are

Findings of the Association for Computational Linguistics: ACL 2023, pages 8529–8537

linguistically informed. Our experiment setup swaps the human learner with the Transformer architecture. Previous studies investigating phonological phenomena with Transformers include Elsner (2021), where Transformers can handle reduplication and gemination. To an extent,[1] the SIGMORPHON shared tasks (Kodner et al., 2022) also demonstrate the capacity of Transformers to represent phonological processes through capturing allomorphs conditioned by phonological environments.

There have been extensive studies on various phonological processes and RNNs. Haley and Wilson (2021) shows that encoder-decoder networks (specifically LSTM and GRU architectures) can learn infixation and reduplication. Mirea and Bicknell (2019) explores whether phonological distinctive feature information is required for learning word-level phonotactic generalisations using LSTMs. The authors find that information about phonological features hinders model performance, and phonotactic patterns are learnable from the distributional characteristics of each segment alone. Moreover, distributional information proves to be integral in recovering phonological categories (Mayer, 2020).

Another way to investigate neural architecture abstractions is to probe the model internally. Silfverberg et al. (2021) examines whether RNN states encode phonological alternations through experiments on Finnish consonant gradation. The authors show that the models often encode consonant gradation in a select number of dimensions. Rodd (1997) probes the hidden states of an RNN model which controls Turkish vowel harmony. Similarly, Silfverberg et al. (2018) establish a correlation between embedding representations and distinctive phonological features for Finnish, Spanish and Turkish. This paper focuses on a model-external interrogation of Transformer generalisations by studying the predictions produced.

3 Language Design

The phonological phenomena in question are tested on a constructed language. The primary motivation for this is to allow for a controlled experiment and ensure that we can generate enough samples of the required phonological environments for rules to be triggered and thus observed. With this in mind, we require the constructed language to be as representative as possible of natural language. Therefore, key features were chosen based on the condition of being the most typologically common ones (Maddieson, 1984; Ladefoged and Maddieson, 1996; Maddieson, 2013). The main characteristics are listed in Table. 1.

Feature	Inventory
Vowel	{a,e,i,o,u}
Consonant	{p,t,k,b,d,g,ʧ,ʤ,f,s,ʃ,v,m,n,ŋ,l,r,w,j}
Onset	{C, Ø, CC}
Nucleus	{V,VV}
Coda	{C, Ø, CC}

Table 1: Phonological inventory and syllable structure of our constructed language. C and V are abstract symbols referring to the full inventory of consonants and vowels, respectively. Where 'VV' occurs in the nucleus, this could be either a diphthong or a long vowel 'V:'.

Generating a lexicon The most complex syllable structure possible in the language is **CCVVCC** and the simplest one is **V**. Since our language design aims to generate a synthetic lexicon, we also control for word length distribution. Previous works have shown that word length over word types exhibits a roughly Gaussian distribution with a mean in the range $[7, 10]$, depending on the language (Smith, 2012). We have chosen a mean word length of 8.

An additional constraint when generating a lexicon is the sonority sequencing principle (SSP) (Selkirk, 1984; Clements, 1990). Syllable structures tend to be highly influenced by the sonority scale, with the general rule that more sonorous elements are internal (i.e., close to the nucleus) and less sonorous elements are closer to the syllable edge. Therefore, we use a sonority metric to avoid generating implausible consonant clusters, with the onset and coda requiring opposite values on the metric, i.e. increasing sonority in the onset and decreasing in the coda.

4 Data[2]

Our data preparation follows three steps: lexicon generation, triplet (`lemma`, `tag`, `surface form`) formation via the finite-state tool *foma* (Hulden, 2009) and, finally, sampling of these triplets ac-

[1] This largely depends on the language considered and the phonological processes it exhibits.

[2] All data and code is available at https://github.com/smuradoglu/phon-proc

cording to the experiment at hand and formatting for Fairseq.(Ott et al., 2019)[3]

We train the model as a standard 'inflection' task (Kodner et al., 2022), but with tags being identifiers of the processes that are to be triggered instead of morphosyntactic information. For example, the input sequence `moupi#GEMINATION` would be paired with the output `mouppi`. More example triplets are shown in Table 2.[4]

Input	Tag	Output
ateiʃa	#APOCOPE	ateiʃ
enpanka	#APHAERESIS	npanka
a:ŋʤ	#SHORTENING	aŋʤ
vepisk	#LENGTHENING	vepi:k
moupi	#GEMINATION	mouppi
aimggi	#DEGEMINATION	aimgi
soute	#INTERVOCALIC	soude
refend	#DEVOICE	refent
ketedu	#METATHESIS	kedetu
totoŋ	#HAPLOLOGY	toŋ
pima	#COPY	pima

Table 2: Sample data showing a subset of phonological phenomena considered. The training/test input data is formatted in triplets: lemma, tag and inflected form. This follows a similar structure as used in the SIGMORPHON shared task for morphological inflection.

Lexicon generation entails generating viable syllable structures and filling these abstract structures using vowel and consonant inventories. The syllables are concatenated n times, where n is an integer between 1 and 10. We sample from this uniform distribution to produce a Gaussian distribution for word length with a mean of 8 symbols.

We include a `COPY` tag, where the input is copied to the output, to negate any performance drop by the model when unseen lemmata are encountered (Liu and Hulden, 2022). In other words, the model, at test time, will never encounter a completely unseen lemma on which to perform a phonological change, since it will always have witnessed at least an input-output pair of any lemma used that is simply copied to the output.

[3]See B for model details.

[4]Our nomenclature of sound changes follows Campbell (2013).

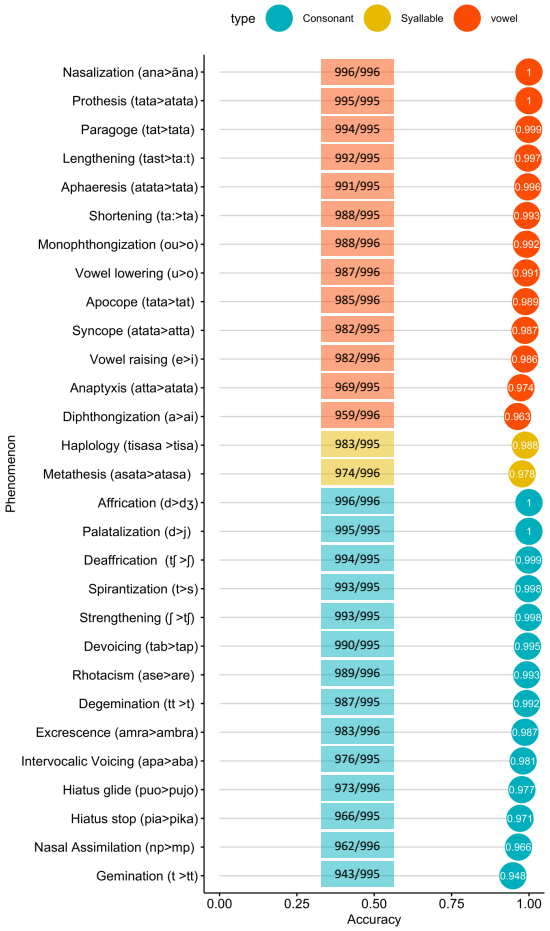

Figure 1: Modelling of Phonological Phenomena. Model accuracy across each phenomena. Labels in bar report details in the following manner: instances of correct prediction/test size. Figures in circle correspond to accuracy.

5 Modelling common phonological processes with varying degrees of complexity

In this experiment, we establish that seq2seq models can successfully capture a range of phonological processes, including more complex rules such as metathesis. As seen in Figure 1, the transformer model performs reasonably well across all phonological phenomena, with little distinction between the complexity of the process considered.

6 Linguistic Category generalisation

We examine whether the transformer model can generalise linguistic categories such as vowels or syllables from examples of alternations. During training, we expose the model to two phenomena at once (priming/held-out cases) of processes where the model could potentially infer relevant

categories and extend this knowledge to withheld cases. The first set of experiments focuses on the generalisation of vowels, and the second centres on categorising consonants.

6.1 Vowel Experiments

6.1.1 Apocope/Aphaeresis

In this experiment, Aphaeresis–deleting word-initial vowels—is the priming process and Apocope—deleting word-final vowels—is the held-out case. The training set consists of aphaeresis cases with all five vowels. In other words, lexicon beginning with **a,e,i,o,u** are included. Apocope examples exclude cases where **u** occurs word-finally. The **u**-final words with the Apocope tag are present only at test time. Table. 3 summarizes the results. From these results, it is clear that the model extends the Apocope rule to the unseen **u**-vowel. There are only 8 instances within the 10 errors where **'u'** is not deleted. The remaining 2 errors are other modelling errors (such as repeating characters): outputting **sou** instead of the gold **so** with input **sou**.

6.1.2 Vowel shortening/lengthening

Following a similar setup to the Apocope/Aphaeresis experiment, the vowel shortening (priming) and lengthening (withheld case **u**) case involves training a model with all vowel cases for shortening, and all vowels except **u** for the vowel lengthening process. The results show a 100% accuracy for the previously unseen **u**-cases for vowel lengthening. The two errors observed are from other categories (i.e., vowel shortening and non-**u** lengthening).

6.2 Consonant Experiments

6.2.1 Gemination/Degemination

This experiment involves training a model for Degemination (priming) and Gemination (withheld case **p**) processes. The results show that the transformer model has successfully extended the consonant category to include the unseen **p**. Out of the 453 test cases, only 12 were incorrect **p** cases, with the remaining five non-target errors. Incorrectly predicted instances follow the pattern of outputting **lup** with input **lup** instead of the gold **lupp**.

6.2.2 Devoicing/Intervocalic voicing

This experiment involves final stop Devoicing (priming) and Intervocalic Voicing (with-held-case **p**). The training set is comprised of all word-final devoicing cases (b>p,d>t,g>k) and all intervocalic cases except the **p** case (where p>b).

Process	Test Size	Accuracy
Aphaeresis	995	0.998
Apocope Overall	1465	0.992
Apocope 'u → 0'	587	0.983
Vowel Shortening	995	0.999
Lengthening Overall	1071	0.999
Lengthening 'u s→ u :'	95	1.000
Degemination	995	0.992
Gemination Overall	1357	0.987
Gemination 'p → p p'	453	0.974
Devoicing	995	1.000
Intervocalic Overall	1196	0.952
Intervocalic 'p→ b'	250	0.776

Table 3: Linguistic Categories Experiment. AA, SL, GD and DI overviews refer to Apocope / Aphaeresis, Shortening / Lengthening, Gemination / Degemination and Devoicing / Intervocalic voicing. The last line refers to the withheld case; e.g. Apocope of **u**.

Process	Test Size	Accuracy
W-Initial voicing	995	1.000
Intervocalic Overall	1196	0.8746
Intervocalic 'p→ b'	250	0.4000
W-Initial devoicing	995	1.000
Intervocalic Overall	1196	0.9473
Intervocalic 'p→ b'	250	0.7480

Table 4: Word initial (de)voicing and intervocalic voicing Experiment. The last line refers to the withheld case; i.e. Intervocalic voicing of **p**.

The results show that **p** is transformed to a **b** 77.6% of the instances. Where the conversion does not take place, errors typically follow the pattern of, e.g. outputting **epeiʃe** instead of **ebeiʃe** with the input **epeiʃe**

To investigate the comparatively low performance. We compare word-initial devoicing with word-initial voicing as a priming process. The results are summarised in Table. 4. The accuracy of the predictions for the unseen **p** was substantially lower in the case of word-initial voicing (40%) compared with the word-initial devoicing (74.8%). Interestingly, word-initial voicing

involves the same process as intervocalic voicing **(p>b)**, with only different environments triggering the process.

7 Word-internal representations

To test whether seq2seq models can learn a representation of word-internal structures, such as syllables, we experiment with examples of haplology. Haplology (**tatasa > tasa**) is the process in which a repeated sequence of sounds is simplified to a single occurrence. For example, if the word **haplology** were to undergo haplology, it would reduce the sequence **lolo** to **lo**, **haplology > haplogy**.

In this experiment, we include two additional processes so the model can witness the contrast between vowels and consonants separately: (1) word-final vowel deletion and (2) word-final consonant deletion.

Process	Test Size	Accuracy
Overview	3264	0.959
→ Consonant deletion	992	0.999
→ Vowel deletion	992	0.998
→ Haplology overview	1280	0.898
Haplology	920	0.972
Unseen CVCV	269	0.944
Double Haplology	91	0.011
VCVC test	2658	0.782

Table 5: Experiment 2: Haplology results. An overview of the experiment is presented, alongside a breakdown for each process. The haplology cases are further split into cases of the unseen CVCV, double haplology (where haplology occurs more than once in a word) and *regular* haplology (which entails words where the haplology rule is applicable and words where it should not be triggered.

To test the generalisation capacity of the model, at test time, we include the following withheld cases: unseen CVCV structures—i.e. cases where haplology should apply, but the specific CVCV-sequence is never seen in the training data; words where haplology occurs more than once; and VCVC structures to see if the model (erroneously) learns to delete any repeating sequence of symbols. In our experiment, we withhold from the training set the following CVCV-sequences: **dede, fofo, kuku, wowo, baba, vivi, papa, titi, soso, momo, nene, rere, lili, ʃuʃu, jiji, ʧuʧu, ŋaŋa, gugu**.

Note that haplology includes both cases where haplology applies and does not since the input word may or may not contain a CVCV-sequence where the two CVs are identical.

Table 7 summarises the results obtained. The model shows high accuracy for the supplementary word-final vowel and consonant deletion processes. We separate the haplology cases further into specific test cases. Our results from the unseen CVCV category show strong evidence for model generalisation of CV structures. We further tested the same model on a separate test set consisting of VCVC structures. We see that for approximately 78% of the set, it correctly recognises these cases as incorrect conditions for haplology. In the remaining instances, the model does show a rare over-generalisation to sometimes delete repeating sequences regardless of the characteristics of the sequence.

The largest source of error within the haplology cases is the scenario in which haplology can be applied twice within the same word. In these cases, typically, the first case of repeating CV is deleted, and the second instance remains untouched, as when outputting **fuejaja** with input **fufuejaja**, instead of the gold **fueja**.

8 Conclusion

The transformer model successfully models all 29 phonological phenomena with slight variation across phenomenon complexity. Our results show that the model can generalize linguistic categories and structures. Through haplology, we show that the model appears to learn to recognize and generalize syllabic structure and is capable of recognizing the difference between VC and CV and can also generalize the transformation triggered by haplology to unseen CV sequences.

Limitations

One drawback of the experiments presented here is the reliance on a constructed language. While we have tried to design a language that is as representative of natural language as possible, there may be additional statistical effects that are not taken into account. For example, it is unlikely that one language would capture all 29 phenomena presented here and that the process would be triggered enough times to produce a large enough corpus. How these findings extended to existing language corpora is an open question for future studies.

References

Lyle Campbell. 2013. *Historical Linguistics*. Edinburgh University Press.

Kyle E Chambers, Kristine H Onishi, and Cynthia Fisher. 2010. A Vowel is a Vowel: Generalizing Newly Learned Phonotactic Constraints to New Contexts. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 36(3):821.

G. N. Clements. 1990. *The Role of the Sonority Cycle in Core Syllabification*, volume 1 of *Papers in Laboratory Phonology*, page 283–333. Cambridge University Press.

Micha Elsner. 2021. What Transfers in Morphological Inflection? Experiments with Analogical Models. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 154–166, Online. Association for Computational Linguistics.

Sara Finley. 2011. Generalization to Novel Consonants in Artificial Grammar Learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33.

Sara Finley and William Badecker. 2009. Artificial Language Learning and Feature-based Generalization. *Journal of memory and language*, 61(3):423–437.

Coleman Haley and Colin Wilson. 2021. Deep Neural Networks Easily Learn Unnatural Infixation and Reduplication Patterns. In *Proceedings of the Society for Computation in Linguistics 2021*, pages 427–433, Online. Association for Computational Linguistics.

Mans Hulden. 2009. Foma: a Finite-State Compiler and Library. In *Proceedings of the Demonstrations Session at EACL 2009*, pages 29–32, Athens, Greece. Association for Computational Linguistics.

Jordan Kodner, Salam Khalifa, Khuyagbaatar Batsuren, Hossep Dolatian, Ryan Cotterell, Faruk Akkus, Antonios Anastasopoulos, Taras Andrushko, Aryaman Arora, Nona Atanalov, Gábor Bella, Elena Budianskaya, Yustinus Ghanggo Ate, Omer Goldman, David Guriel, Simon Guriel, Silvia Guriel-Agiashvili, Witold Kieraś, Andrew Krizhanovsky, Natalia Krizhanovsky, Igor Marchenko, Magdalena Markowska, Polina Mashkovtseva, Maria Nepomniashchaya, Daria Rodionova, Karina Scheifer, Alexandra Sorova, Anastasia Yemelina, Jeremiah Young, and Ekaterina Vylomova. 2022. SIGMORPHON–UniMorph 2022 Shared task 0: Generalization and Typologically Diverse Morphological Inflection. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 176–203, Seattle, Washington. Association for Computational Linguistics.

Peter Ladefoged and Ian Maddieson. 1996. *The Sounds of the World's Languages*, volume 1012. Blackwell Oxford.

Ling Liu and Mans Hulden. 2022. Can a Transformer Pass the Wug Test? Tuning Copying Bias in Neural Morphological Inflection Models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 739–749, Dublin, Ireland. Association for Computational Linguistics.

Maddieson. 1984. *Patterns of Sounds*. Cambridge Studies in Speech Science and Communication. Cambridge University Press.

Ian Maddieson. 2013. Syllable structure. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Connor Mayer. 2020. An Algorithm for Learning Phonological Classes from Distributional Similarity. *Phonology*, 37(1):91–131.

Nicole Mirea and Klinton 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*, pages 1595–1605, Florence, Italy. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Jennifer Rodd. 1997. Recurrent Neural-Network Learning of Phonological Regularities in Turkish. In *CoNLL97: Computational Natural Language Learning*.

Elisabeth Selkirk. 1984. 0.(1982). The Syllable. *The Structure of Phonological Representations, Part I, Foris, Dordrecht*, pages 337–382.

Miikka Silfverberg, Lingshuang Jack Mao, and Mans Hulden. 2018. Sound Analogies with Phoneme Embeddings. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2018*, pages 136–144.

Miikka Silfverberg, Francis Tyers, Garrett Nicolai, and Mans Hulden. 2021. Do RNN States Encode Abstract Phonological Alternations? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5501–5513, Online. Association for Computational Linguistics.

Katrin Skoruppa and Sharon Peperkamp. 2011. Adaptation to Novel Accents: Feature-based Learning of Context-sensitive Phonological Regularities. *Cognitive Science*, 35(2):348–366.

Reginald D Smith. 2012. Distinct Word Length Frequencies: Distributions and Symbol Entropies. *arXiv preprint arXiv:1207.2334*.

A Summary of phonological processes

Affrication a process where either a stop, or fricative, becomes an affricate.

Anaptyxis (VCCV > VCVCV) a kind of epenthesis where an extra vowel is inserted between two consonants.

Aphaeresis (atata > tata) the deletion of word initial vowels.

Apocope (tata > tat) the loss of a sound, usually a vowel, at the end of a word.

Deaffrication an affricate becomes a fricative.

Degemination (CC > C) a sequence of two identical consonants is reduced to a single occurrence.

Devoicing the devoicing of stops word-finally.

Diphthongization an original single vowel changes into a sequence of two vowels.

Excrescence (amra > ambra; anra > andra; ansa > antsa) the insertion of a consonant. In our case, the insertion of **b**, **d**, or **t**.

Gemination (C > CC) produces a sequence of two identical consonants from a single consonant.

Hiatus glide (puo > pujo) a semi-vowel/glide is inserted between falling vowel pair.

Hiatus stop (pia -> pika) the insertion of a stop which breaks up a falling vowel pair.

Intervocalic Voicing various sounds become voiced between vowels, in this case we focus on stops.

Lengthening (tast > ta:t) a vowel lengthens subsequent to the loss of a following consonant, also called *compensatory lengthening*.

Metathesis (asta > atsa; asata > atasa) a change in which sounds exchange positions with one another within a word.

Monophthongization a diphthong changes into a single vowel.

Nasal Assimilation (np > mp) the change of nasal sounds to agree with the place of articulation of following stops.

Nasalization (ana > ãna) vowels become nasalized before nasal consonants.

Palatalization (k -> ʧ, or d -> j) involves the change of a velar/alveolar sound to palato-alveolar, this often takes place before or after i or e.

Paragoge (tat > tata) adds a vowel to the end of a word.

Prothesis (tata > atata) a kind of epenthesis in which a sound is inserted at the beginning of a word.

Rhotacism (ase > are) s becomes r; this takes place between vowels or glides.

Shortening (ta: -> ta) vowels shorten in a variety of contexts, e.g. word-finally.

Spirantization an affricate is weakened to a fricative, or a stop to a fricative.

Strengthening fortition of sounds; an affricate becomes a stop, or a fricative becomes an affricate.

Syncope (atata > atta) the loss of a vowel from the interior of a word (not initially or finally)

Vowel lowering results in high vowels becoming mid or low vowels, or mid vowels becoming low.

Vowel raising is where low vowels raise to mid (or high) vowels, or mid vowels to high vowels).

B Model details

Hyperparameter	Value
Encoder/Decoder layers	4
Encoder/Decoder attention heads	4
Optimization	Adam
Embedding size	256
Hidden layer size	1024
Learning rate	0.001
Batch Size	400
Label Smoothing	0.1
Gradient clip threshold	1.0
Warmup updates	1000
Max updates	6000

ACL 2023 Responsible NLP Checklist

A For every submission:

☐ A1. Did you describe the limitations of your work?
Left blank.

☐ A2. Did you discuss any potential risks of your work?
Left blank.

☐ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☐ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☐ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
Left blank.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Left blank.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☐ Did you run computational experiments?

Left blank.

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☐ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Left blank.

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Left blank.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Left blank.

D ☐ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Left blank.