Frequency matters: Modeling irregular morphological patterns in Spanish with Transformers

Anonymous ACL submission

Abstract

The present paper evaluates the learning behaviour of a transformer-based neural network with regard to an irregular inflectional paradigm. We apply the paradigm cell filling problem to irregular patterns. We approach this problem using the morphological reinflection task and model it as a character sequence-to-sequence learning problem. The test case under investigation are irregular verbs in Spanish. In addition to regular verbs, certain verbs follow an irregular L-shaped pattern where the first-person singular present indicative stem matches the stem used throughout the present subjunctive. We examine the role of frequency during learning and compare models under differing input frequency conditions. We train the model on a corpus of Spanish with a realistic distribution of regular and irregular verbs to compare it with models trained on input with augmented distributions of (ir)regular words. We explore how the neural models learn this L-shaped pattern using post-hoc analyses. Our experiments show that, across frequency conditions, the models are surprisingly capable of learning the irregular pattern. Furthermore, our post-hoc analyses reveal the possible sources of errors. All code and data are available at `https://anonymous.4open.science/r/modeling_spanish_acl-7567/` under MIT license.

1 Introduction

A common generation task in morphology is morphological inflection, which is to generate an inflected form for a given target feature tag and the corresponding lemma, e.g., `(lemma:decir, target tag:<V;IND;PRS;1;SG>)` ↦ `digo`. A central challenge in understanding how speakers handle morphological inflection is the Paradigm Cell Filling Problem (PCFP) (Ackerman et al., 2009), which asks how speakers are able to reliably produce inflected forms of words (lexemes) they have never encountered before, even for irregular patterns in the language. Our study focuses on applying PCFP to morphomic patterns, an irregular verb pattern particularly from Romance languages (Herce, 2020). The morphomic pattern, as introduced by Spencer and Aronoff (1994), is a morphological pattern that exist independently of semantics or syntax. They are purely based on the form and structure of words. A key characteristic of morphomic patterns is their predictability within the verbal paradigm, which arises from the consistent sharing of morphological features among certain forms within a verb paradigm, despite the lack of apparent semantic or syntactic motivation (Blevins, 2016; Maiden, 2018). Maiden (2011, 2018, 2021) identified morphomic patterns across Romance languages, highlighting their predictability and systematic nature. An example is the Spanish verb *decir* 'to say' (see Table 1). The forms "digo" (1st person singular, indicative) and "digan" (3rd person plural, subjunctive) share a stem "dig-". This shared morphological feature is part of a morphomic pattern. However, there is no obvious semantic or syntactic property that links "digo" and "digan" while excluding "digo" and "dicen" (3rd person plural, indicative), which uses a different stem "dic-". The pattern affects the Spanish *er* and *ir* verb conjugation classes. In Spanish verbs, all stem alternation patterns have been argued to be morphomic in nature (Maiden, 2018). Although there are various morphomic patterns (such as N-shaped and P-shaped) in Spanish (Maiden, 2018; Herce and Allassonnière-Tang, 2024), the focus of this present study will be specifically on verbs that exhibit the L-shaped morphomic pattern in their inflectional paradigms. Our choice of this pattern is motivated by prior studies that examined its learnability by human (Nevins et al., 2015; Cappellaro et al., 2024). The L-shaped morphome involves a distinct stem form appearing in the first person singular present indicative and all cells of the present

subjunctive mood. As a result, the irregular verb *decir* exhibits the L-shaped morphome pattern, as shown in Table 1.

'to say'	***Indicative***		***Subjunctive***	
	Orthographic	IPA	Orthographic	IPA
1SG	digo	dˈigo	diga	dˈiga
2SG	dices	dˈises	digas	dˈigas
3SG	dice	dˈise	diga	dˈiga
1PL	decimos	desˈimos	digamos	digˈamos
2PL	decís	desˈis	digáis	digˈajs
3PL	dicen	dˈisen	digan	dˈigan

Table 1: A Spanish example of the Romance L-pattern, verb *decir* 'to say'. L-shaped pattern cells are shaded.

Spanish L-shaped verbs are found in relatively few word types but have high token frequency (Maiden, 2011). It has been argued that type frequency, which refers to the number of different words that follow a particular morphological pattern, is the basis for the productivity of morphological patterns (Bybee, 1995; Pierrehumbert, 2001; Bybee, 2003; Albright and Hayes, 2003; Baer-Henney and van de Vijver, 2012; del Prado Martın et al., 2004). Productivity in this context means the ability of a morphological pattern to be used to create new word forms. The present study focuses on the role of type frequency in modeling the acquisition of irregular morphological patterns. Specifically, we investigate the role of type frequency in the learnability of morphomic patterns through a morphological reinflection task as a PCFP in a transformer-based neural network. For the reason of cognitive plausibility, in this work, we opted for the morphological reinflection task. To address this challenge, we employ a multi-source setup of the morphological reinflection task (Kann et al., 2017), which uses multiple source form-tag pairs instead of one form-tag pair. The task involves generating an inflected form from two source form-tag pairs and the target feature tag to predict the target inflected form, e.g., (`source form:digas, source tag:<V;SBJV;PRS;2;SG>, target tag:<V;IND;PRS;1;SG>`) $\mapsto$ `digo`. The reinflection task is particularly challenging due to the variability of the starting point (the source), which can be any of the other inflected forms of the same lemma. This variability makes the task more cognitively plausible, as it reflects the data sparsity encountered by human speakers, who never encounter all of the inflected forms of their language (Blevins et al., 2017). Furthermore, this task aligns with the morphological framework of abstraction based on data directly available to speakers (i.e., inflection forms) (Blevins, 2006; Boyé and Schalchli, 2019), providing a more realistic model of language acquisition and processing. The choice of the task is motivated by the fact that a learner can only determine whether or not a verb is L-shaped or not by knowing at least two of the paradigm cells, specifically one cell from the L-shaped pattern cells and another cell that is not one of the L-shaped pattern cells (see Table 1). Moreover, previous studies have shown that using two source form-tag pairs is sufficient for achieving high accuracy, with no additional improvement from having more than two sources (Silfverberg and Hulden, 2018a; Liu and Hulden, 2020). This task was also successfully employed to examine the learnability of the L-shaped pattern in human (Nevins et al., 2015; Cappellaro et al., 2024). The main aims of the study are: 1) To model the learning of the irregular pattern in Spanish using transformer neural networks trained on data with varying frequencies of irregular vs. regular verbs. 2) To analyze the models' performance and error patterns to get insights into how input frequencies and irregularity impact learning of morphomic patterns.

2 Related Work

Early non-neural approaches to morphological reinflection focused on learning string edit rules from data using sequence-to-sequence models (Albright, 2002; Durrett and DeNero, 2013) and transductions (Nicolai et al., 2015). Finite-state methods were also employed, such as extracting inflections from paradigms using finite-state constructions (Ahlberg et al., 2015) and modeling the mapping between lemmas and inflected forms using Weighted Finite-State Transducers (Alegria and Etxeberria, 2016). Other studies developed morphological analyzers that learn from training data and apply the learned patterns to re-inflect test data (Taji et al., 2016). Discriminative transducers were also used to search for character transformation rules to perform inflection (Nicolai et al., 2016). Additionally, systems based on affixing rules were applied to generate inflected forms by appending or altering affixes based on morphological rules (Liu and Mao, 2016; Cotterell et al., 2017). The introduction of neural-based sequence-to-sequence models marked a significant milestone in modeling morphological reinflection (Kann and Schütze, 2016; Malouf, 2016; Faruqui et al., 2016). Building on this ap-

proach, hard monotonic attention models were introduced to enforce strict alignment between input and output sequences (Wu and Cotterell, 2019). Multi-source setups were also proposed using bidirectional Long Short-Term Memory networks (Kann and Schütze, 2016). Recent advancements include the use of character-level Long Short-Term Memory networks (Silfverberg and Hulden, 2018b) and encoder-decoder based transformers (Wu et al., 2021). Phonologically-aware embeddings have also been proposed to capture both orthographic and phonetic information of words (Guriel et al., 2023). Alternative approaches include treating morphological reinflection as a classification problem (Shcherbakov and Vylomova, 2023) and using imitation learning (Makarov and Clematide, 2018). Finally, Large Language Models have been explored for morphological reinflection, including the analysis of ChatGPT's capability in morphological generation across multiple languages (Weissweiler et al., 2023).

3 Methodology

Model Description This study focuses on the learning of a Transformer model (Vaswani et al., 2017) for morphological inflection tasks. Despite having fewer parameters, the Transformer has been shown to outperform recurrent sequence-to-sequence baselines on morphological inflection tasks (Wu et al., 2021). Specifically, character-level encoder-decoder neural models with attention have achieved high performance on inflectional tasks across many languages (Cotterell et al., 2017, 2018; Vylomova et al., 2020). Our model is based on the encoder-decoder Transformer for character-level transduction proposed by Wu et al. (2021). We implement the model using Fairseq (Ott et al., 2019), a PyTorch-based sequence modeling toolkit. The model consists of four layers with four attention heads, an embedding size of 256, and a hidden layer size of 1,024. We use the Adam Optimizer (Kingma and Ba, 2015) with an initial learning rate of 0.001, a batch size of 400, 0.1 label smoothing, and a 1.0 gradient clip threshold. The model is trained for a maximum of 10,000 optimizer updates, with checkpoints saved every ten epochs. Beam search is used at decoding time with a beam width of five.

Data construction We use the Spanish dataset from the Universal Morphology (UniMorph) project[1] that contains information on Spanish verbs. Entries are coded in the Unimorph scheme (Sylak-Glassman, 2016). For instance, the `V;IND;PRS;1;SG` label for a first person singular present tense verb form, such as `digo`, is decomposed into [POS=VERB, mood=INDICATIVE, tense=PRESENT, person=1, number=SINGULAR]. We describe the lemma, the inflected word (both in IPA), and a bundle of morphological features as follows: (`lemma, form, feature`). We refer to the verbs as `lemmas`; the verb forms as `forms`; the morphosyntactic description (MSD) as `features`. An example entry is (`desiɾ, digo, V;IND;PRS;1;SG`). For training, we then convert these entries to two source form-tag pairs, the target feature tag and the target inflected form. For example, let us say the first source form is `digo` (1st person singular, indicative), the second source form is diga (1st person singular, subjunctive) and the target form is digas (2nd person singular, subjunctive), then the above entry is converted to a so-called *triple* entry such as (`digo, V;IND;PRS;1;SG, diga, V;SBJV;PRS;1;SG, V;SBJV;PRS;2;SG, digas`). The dataset contains 5,460 distinct lemmas, of which 300 are L-shaped lemmas, and 4,860 are NL-shaped lemmas. 382,956 triples were formed. To investigate the role of the input frequencies, we created augmented corpora by manipulating the training input to use them under different training conditions. To operationalize the factor of input frequency in our experiments, three experimental conditions were created by varying the relative amount of L vs. NL-shaped verbs used in training. We investigate model behaviour under a) a naturalistic frequency distribution in our first condition with 10% L-shaped verbs and 90% NL-shaped verbs (henceforth, *10%L-90%NL condition*) to reflect a realistic frequency distribution of the Spanish language[2] and two counterfactual conditions with an increase in the frequency of L-shaped verbs, and a decrease in the frequency of the NL-shaped verbs: b) 50% L-shaped verbs and 50% NL-shaped verbs (henceforth, *50%L-50%NL condition*) and c) 90% L-shaped verbs and 10% NL-shaped verbs (henceforth, *90%L-10%NL condition*). The relative frequency of these counterfactual conditions was created to allow a direct comparison of the learnability of L-shaped verbs relative to NL-shaped verbs. The 50%L-50%NL

[1] https://unimorph.github.io/

[2] This is similar to the relative frequencies of L and NL-shaped verbs in the dataset which is 6% L and 94% NL.

condition contains the same amount of L-shaped verbs and NL-shaped verbs. The 90%L-10%NL condition contains the exact opposite of the first condition with 90% L-shaped verbs and 10% NL-shaped verbs.

Data representation The input sequence consists of space separated characters of the first source form followed by # and the angle bracket delimiters for the feature tag, followed by # and space separated characters of the second source form with # and the angle bracket delimiters for the feature tag and finally with the angle bracket delimiter for the feature tag of the target form. An example input sequence is:

`d i g a # <V;SBJV;PRS;1;SG> # d i g a s # <V;SBJV;PRS;2;SG> # <V;IND;PRS;1;SG>`

The expected target output based on the ground truth would be the space separated characters forming the target word, `d i g o`. We refer to this input-output sequence as a *combination*.

Data sampling In order to examine the effect of the *relative* (as opposed to *absolute*) frequency of L vs. NL-shaped verb, the total number of distinct lemmas was fixed across these conditions. As mentioned above, only 300 L-shaped lemmas were found in the dataset, therefore, the maximum number of L-shaped lemmas in the 90%L-10%NL condition is capped to 300, which represents 90%L. Therefore, the training set should contain 333 lemmas, and we need to sample 33 NL-shaped verbs (amongst the 4,860 NL-shaped verbs) to represent 10% of NL. Similarly, we sampled for the other two conditions (50%L-50%NL and 10%L-90%NL), see Figure 1.

It has been shown that if the forms of a lemma exist in both the training phase and the test phase (i.e., there is lemma overlap), the models' performance would be artificially inflated (Goldman et al., 2022). Building on this insight, we conducted the train-dev-test splits at three levels: *lemma level*, *combination level* and *run level*. At the *lemma level*, we ensure that there is no lemma overlap between the training, the development and the test splits. Given that each lemma produces around 600 combinations and there are 333 lemmas in each condition, 199,800 combinations are generated. We decided against using the entirety of these combinations in a single model due to reasons of cognitive implausibility and computational infeasibility. At the *combination level*, we therefore, divide the 333 lemmas (in the form of combinations) into four bins, each retaining the same distribution of L and NL-shaped lemmas of the specific condition (e.g., 10%L-90%NL). In each combination bin, the lemmas are split into 70% for training, 10% for development, and 20% for testing. Finally, to assess the robustness of the model, at the *run level*, we shuffle the order of the combinations used in training three times since the order of input might matter during training. Thereby, we obtain three runs for each combination bin.

The training data includes full inflection tables, and the trained model is to task to inflect the unseen lemmas (i.e., not seen in training). For the development and test data, every two-slot combination of given slots is used as input to predict the target form corresponding to the target MSD tag. For all frequency conditions, this data sampling procedure results in a dataset with a training set of 39,435 combinations, development set of 4,455 combinations, and a test set of 44,220 combinations. The test set remains the same for each combination bin, which helps in identify potential overfitting issues. In total, we get 12 such datasets for each condition. See Figure 2 for an illustration of our data sampling procedure.

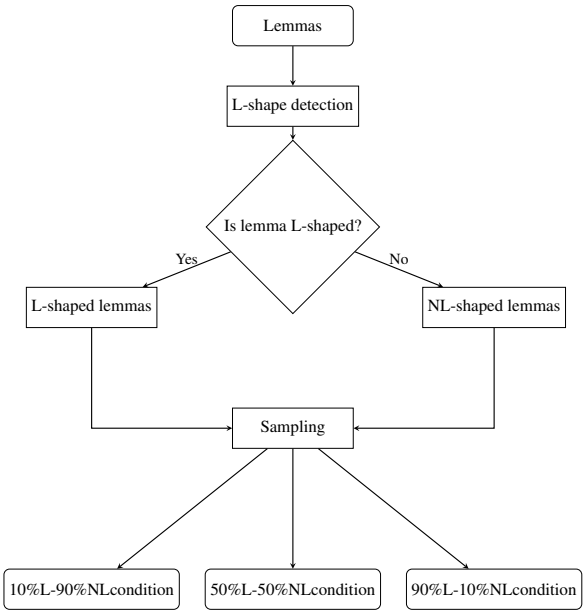

Figure 1: Flowchart showing the process for creating the datasets for the three frequency conditions (10%L-90%NL, 50%L-50%NL, and 90%L-10%NL) at the *lemma level*.

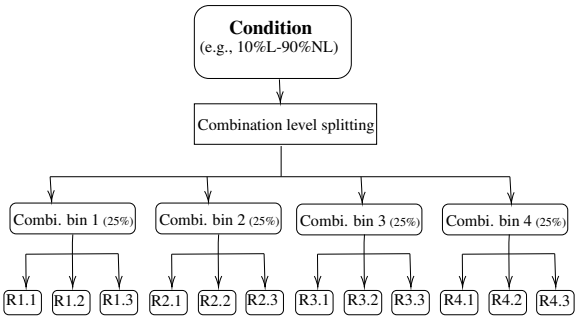

Figure 2: Flowchart showing the process for creating the dataset for each condition at the *combination level*, and *run level*.

4 Analysis and Results

We first present the overall sequence accuracies of the models in the three conditions before we turn to more detailed error analyses in post-hoc analyses. In the post-hoc analyses, we aim specifically at understanding the factors that influence the learning of irregular verbal paradigms. We systematically investigate the models' performance in the three frequency conditions on L-shaped verbs by first exploring possible primacy/recency effects (Wang et al., 2023; Liu et al., 2024) in Section 4.1. Second, we investigate the models' sensitivity to effects of memorization/generalization (Hupkes et al., 2023) in Section 4.2, and finally, we examine the models' behaviour for specific phonological alternations in Section 4.3.

In the morphological inflection task, errors can occur in the suffixes and the stem of the predicted form (Kodner and Khalifa, 2022). We evaluate two types of accuracies: sequence and stem-only accuracies.

First, we evaluate the sequence accuracies of L-shaped and NL-shaped verbs across frequency conditions. To examine the impact of verb type distribution (L-shaped vs. NL-shaped) on model performance, sequence accuracies were calculated separately for L-shaped and NL-shaped verbs across the three conditions, see Figure 3. The results show that the majority verb type in each condition achieved higher accuracies. In the 10%L-90%NL condition, NL-shaped verbs had a mean accuracy of 61.8%, but L-shaped verbs had a mean accuracy of 36.75%. Conversely, in the 90%L-10%NL condition, L-shaped verbs had a mean accuracy of 88.75%, but NL-shaped verbs had a mean accuracy of 24.17%. In the 50%L-50%NL condition, L-shaped verbs had a mean accuracy of 72.31%, and NL-shaped verbs had a mean accuracy of 55.19%.

Next, we analysed the stem accuracies. The results show that L-shaped verbs had higher stem accuracies when they were more prevalent in the training data. In the 10%L-90%NL condition, NL-shaped verbs had a mean stem accuracy of 68.59%, while L-shaped verbs had a mean stem accuracy of 56.7%. In the 50%L-50%NL condition, L-shaped verbs had a mean stem accuracy of 77.24%, while NL-shaped verbs had a mean stem accuracy of 70.25%. In the 90%L-10%NL condition, L-shaped verbs had a mean stem accuracy of 89.51%, while NL-shaped verbs had a mean stem accuracy of 31.09%, illustrated in Appendix A.1. This shows the performance difference of the models based on the distribution of verb types in the training data. Overall, we found an expected improvement in learning with a higher frequency in the input. The neural network performed better on NL-shaped verbs than on L-shaped verbs in the 10%L-90%NL condition, which is the most naturalistic condition. However, we also found an overall learning advantage for L-shaped verbs compared to NL-shaped verbs in other conditions, suggesting that the neural network might be learning specific characteristics of L-shaped verbs that give them an edge. The following sections report on post-hoc analyses conducted to further understand how (ir)regularity determine the learning of verbs in the transformer.

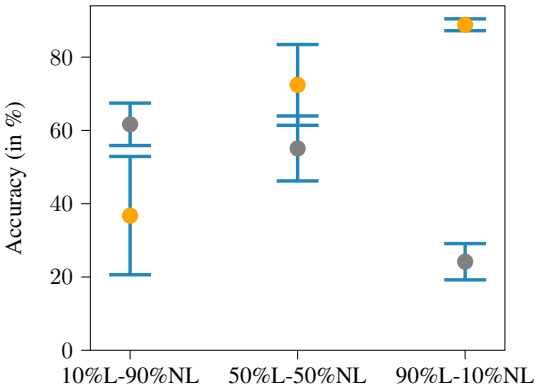

Figure 3: Mean and confidence intervals of overall sequence accuracies for L-shaped and NL-shaped verbs across 10L-90NL, 50L-50NL and 90L-10NL conditions. Gray: NL-shaped, Orange: L-shaped.

4.1 Cell combinations

In this section, we investigated how the combination of paradigm cells influences the learning of L-shaped verbs and explore whether certain cell characteristics lead to better learning outcomes. The analysis of the cell combinations shows clear

evidence of the primacy effect but no evidence of a clear recency effect across conditions: Each cell combination consists of three parts: the source 1 cell , the source 2 cell, and the target cell. Cells within the L-shaped morphomic pattern are labeled as *In*, and those outside are labeled as *Out*. For example, a combination with source 1 as digo (In), source 2 as dices (Out), and target form as diga (In) is categorized as In-Out-In.

	10%L-90%NL			50%L-50%NL			90%L-10%NL		
Cell combi.	L (%)	NL (%)	L/NL	L (%)	NL (%)	L/NL	L (%)	NL (%)	L/NL
In-In-In	45.61	60.64	0.7	82.56	53.75	1.63	90.02	24.14	6.85
In-Out-Out	4.44	34.84	0.08	57.18	34.58	1.78	91.53	40.66	2.86
In-In-Out	27.17	62.57	0.4	63.94	54.19	1.15	87.66	22.85	5.07
In-Out-In	40.26	45.4	0.9	80.47	45.83	1.92	91.98	48.72	6.07
Out-In-In	40.62	64.77	0.57	76.17	57.19	1.41	89.54	16.93	15.23
Out-In-Out	37.68	66.68	0.54	70.49	59.58	1.18	87.43	16.53	6.58
Out-Out-In	30.39	59.23	0.45	59.8	52.11	1.08	87.06	22.42	7.44
Out-Out-Out	25.56	58.87	0.38	69.77	56.19	1.23	86.7	18.4	6.19

Table 2: Cell combination accuracies for 10L-90NL (left), 50L-50NL (middle), and 90L-10NL (right). The mean accuracies in percentage are calculated for separately by verb types (L denotes L-shaped verbs and NL denotes NL-shaped verbs) and by cell combinations (e.g., In-In-In). L/NL denotes the ratio of the mean accuracies of the L-shaped vs NL-shaped verbs. For a visualization of this table, see Appendix A.2.

First of all, the main frequency effect is consistently found across all eight cell combinations (In-In-In, In-Out-Out, etc.), see Table 2. We then explore the primacy/recency effect by comparing pairs of cell combinations to determine if the target cell aligns with either source 1 (primacy) or source 2 (recency), e.g., **In**-Out-**In** allows for a primacy effect, while Out-**In**-**In** allows for a recency effect. To evaluate if there is a recency effect, we identify minimally different pairs of cell combination, e.g., In-Out-In (primacy) can be compared to Out-Out-In for the presence of a primacy effect. A consistent primacy effect is observed for L-shaped verbs in both 10%L-90%NL and 50%L-50%NL conditions, though it is weaker in the 90%L-10%NL condition. For e.g., in the 10%L-90%NL, the accuracy of In-Out-In cell combination (40.26%) is greater than Out-Out-In (30.39%), indicating the presence of a primacy effect. No consistent recency effect is detected across conditions. Next, the analysis shows no strong pattern favoring mixed (In-Out-Out, etc.) or non-mixed (In-In-In, Out-Out-Out) combinations. Finally, we find that 'In' targets are predicted more accurately than 'Out' targets for L-shaped verbs. For example, in the 10%L-90%NL condition, the accuracy of Out-In-In (40.62) cell combination is greater than In-In-Out (27.17).

4.2 Memorization and Generalization

We examine the models' ability to memorize and generalize morphomic patterns for *stem-final consonant triples*. Specifically, we ask, to what extent the model performs in seen vs. unseen patterns across frequency conditions. Stem-final consonants are those that appear at the end of a verb stem, and we focus on triples formed by the stem-final consonants of the first source form, second source form, and target form. For example, given the forms (ʈrad'usen (source 1), ʈradusk'amos (source 2), ʈrad'uskan (target)) , the stem-final consonant triple consists of s, sk, and sk. Recent research (Hupkes et al., 2023) shows that sequence-to-sequence models apply learned linguistic patterns to seen triples; however, applying learned patterns to unseen lemmas remains a challenging task due to the model's limited capacity for encoding complex structures. We thus investigate two types of knowledge state: Memorization and Generalization. Memorization is the model's ability to apply a pattern to seen triples, operationalized as the model's ability to predict stem-final consonant triples present in the training data, calculated as the ratio of correctly predicted seen triples to total seen triples. Generalization is the model's ability to apply a pattern to unseen triples, operationalized as the model's capability to predict triples not present in the training data, calculated as the ratio of correctly predicted unseen triples to total seen triples. We evaluated the models' performance on L-shaped forms depending on frequency conditions (10%L-90%NL vs. 50%L-50%NL vs. 90%L-10%NL), and knowledge state conditions (memorization vs. generalization). Logistic mixed-effects models are employed using the `glmer` function from the `lme4` package in `R`. The model predicts accuracy (`prediction_status` (correct vs. incorrect) with two fixed effects, frequency conditions and knowledge state conditions, and two random intercepts `triples` and `model` (among 12 models).

```
glmer(prediction_status ~
knowledge_state * frequency_condition +
(1|triples) + (1|model),  data=df,
family="binomial")
```

To interpret the results, the emmeans package (Lenth, 2024) is used to calculate estimated marginal means (EMMs) from the fitted generalized logistic mixed model. When the frequency distribution of verb types is equal, the predicted proba-

bilities in generalized forms are slightly higher than in memorized forms (0.113), in other frequency conditions, predicted probabilities are higher in memorized forms (10%L-90%NL: 0.036; 90%L-10%NL: 0.304), see Appendix A.3 for details. For memorization, we find that when the frequency of L-shaped verbs increases across frequency conditions, the odds of correct predictions also increase. For generalization, we find that the odds of a correct prediction are highest when the training consisted of equal distribution of verb types. The odds of correct predictions decrease when training data has a majority and minority pattern. Overall, model performance on memorization and generalization tasks varies across conditions and we observe the expected advantage for memorized forms over generalized ones.

4.3 Consonant-pair analysis

This section shows that models are sensitive to the input frequency of alternations, indicating that they have not fully learnt the abstract morphological patterns. We explore the models' performance on specific stem-final consonant pairs of L-shaped verbs. These alternating pairs consist of the stem-final consonant of the forms in the out cells and that of the in cells of the paradigm (see section 4.1). For example, for the lemma /desiɾ/, the consonant pair is [s]-[g], where [s] is the stem-final consonant of the out cells and [g] is found in forms sharing the L-shaped pattern. The most frequent pairs in the dataset are [s]-[sk], with 141 occurrences, followed by [n]-[ng] and [ç]-[x], with 53 and 25 occurrences, respectively. This unbalanced distribution naturally leads to differing ratios for each run due to the process of data sampling and may lead to more productive generalization for frequent alternations. We examine these pairs' frequency in training and test datasets across the frequency conditions to assess how L-shaped verb proportions affect learning. Across all 3 runs of the 10%L-90%NL condition, [s]-[sk] appears frequently in both test (3-4 times) and training sets (8-13 times). Some pairs like [lç]-[lx] and [s]-[g] appear in test sets but are rare or absent in training, which might pose difficulty for models in applying patterns to novel combinations. In the 50%L-50%NL condition, [s]-[sk] remains the most frequent pair, appearing 14-22 times in test sets and 15-23 times in training sets. Other pairs like [n]-[ng] and [s]-[g] also appear but less frequently. In the 90%L-10%NL condition, [s]-[sk] appears even more frequently (22-32 times in test sets and 97-104 times in training), while other pairs remain less common. Details are given in Appendix A.4. A confusion matrix in Appendix A.5 summarizes the top 5 most erroneous consonant-pairs for L-shaped verbs. The main frequency effect can still be found consistently across these consonant-pairs, with their accuracies increase with an increase in L-shaped verb proportions. In the 10%L-90%NL condition, [s]-[sk] achieves 68.6% accuracy, while [s]-[g] and [n]-[ng] reach 26.2% and 77.4%, respectively. Accuracy increases in the 50%L-50%NL condition to 89.8%, 60.6%, and 83.6%, respectively. In the 90%L-10%NL condition, accuracies further improve to 93.3%, 92.6%, and 91.4%. Looking beyond accuracies, we found that the models still make systematic errors, often defaulting to more frequent lemma consonants rather than altered ones (e.g., predicting [s]-[s] instead of [s]-[sk]). Models perform worse for less frequent or unseen alternations (such as [s]-[g]), compared to more frequent alternations ([s]-[sk]). This indicates that the models did not learn the abstract morphological rules well. In these cases, models tend to regularize L-shaped verbs in datasets, as erroneous predictions often result in non-alternating pairs. These findings highlight the challenges models face in generalizing less common patterns despite improved performance with increased exposure to dominant patterns.

5 Conclusion and future work

We set out to address whether transformer-based models can learn an irregular pattern in Spanish trained on augmented corpora with varying frequencies of irregular and regular verbs. Overall, we observed an advantage for L-shaped over NL-shaped verbs, especially when comparing the uneven frequency conditions where the minority verb type had lower accuracy. These results show that the transformer models' can learn to inflect regular and irregular Spanish verbs when trained on datasets with varying proportions of regular and irregular verbs. The neural network model's ability to learn irregular patterns (morphomes) aligns with prior research on related language tasks, such as English past tense inflection (Rumelhart et al., 1986; Ellis and Schmidt, 1998; Cotterell et al., 2016) and SIGMORPHON shared tasks on morphological inflection across multiple languages (Cotterell et al., 2017, 2018).

We analyzed the error patterns to gain insights into how input frequencies and irregularity impact the learning of morphomic patterns in transformer-based neural networks. Our analyses reveal that these models are sensitive to the frequency of irregular verbs in the training data. Also, we investigated how the combination of paradigm cells influences the learning of L-shaped verbs and found a primacy effect, indicating that cells presented earlier in the sequence are learned more effectively. We did not observe a consistent recency effect, suggesting that the models do not overly rely on cells presented later in the sequence. Next, we demonstrated how well the model can apply learned morphomic patterns to new data. As expected, the accuracy is better on memorized forms than on generalized forms. However, we did not find a consistent generalisation effect, such that the model's ability to generalize was best observed in the 10%L-90%NL condition. We found a clear memorization effect, and it becomes increasingly important as the proportion of L-shaped verb increases. We also examined the frequency of specific consonant pairs in the training and test datasets across different frequency conditions and analyzed the models' performance in predicting these pairs. The analysis shows that certain consonant pairsare consistently frequent across different conditions, and others have varying frequency. The model tends to regularize L-shaped verbs if their alternating consonant pairs are infrequent or unseen in the training data.

Our findings motivate a comparison of the behaviour of a transformer-based neural network with human behaviour. Specifically, we aim to compare our model's performance in a wug-test-like inflection task with the experimental results from human speakers, such as those from Nevins et al. (2015) and Cappellaro et al. (2024). This comparative analysis can reveal how well our computational approach captures the linguistic and cognitive phenomena observed in these two seminal human studies. Finally, there is a debate on whether incorporating prior linguistic information, such as morphological rules or phonological constraints, into character-level models is necessary Ling et al. (2015); Chung et al. (2016). Exploring the impact of incorporating such linguistic information into transformer-based models could enhance their ability to capture morphomic patterns more effectively.

Limitations

We have not explored the impact of different hyperparameter settings on the model's performance. Adjusting these hyperparameters can significantly affect the model's ability to learn(Wiemerslage et al., 2024). We have not used other computational approaches. For instance, Recurrent Neural Networks (RNNs) (as used in the study for modeling German plurals (Dankers et al., 2021)), the use of Linear Discriminative Learning Model (Baayen et al., 2019; Heitmeier et al., 2021; Jeong et al., 2023) or Deep Discriminative Learning Model (Heitmeier et al., 2024). To rigorously evaluate the model's generalization capabilities, it would be beneficial to use test data that is entirely unattested in the training set. This means that both the lemmas (word stems) and the feature tags (e.g., tense, number, person) should be novel to the model. This approach, similar to Kodner et al. (2023), would provide a more robust assessment of the model's ability to generalize morphomic patterns to unseen data.

We have not taken into account the morphological complexity of the verbs. Some verbs have prefixes, and some do not, therefore,k some lemmas share the same stems. On the one hand, this renders the lemma-level train-development-test splitting procedure less effective and might artificially inflate the accuracies of our models. On the other hand, this is arguably ecologically more valid as human learners do get exposed to both morphologically complex and simple verbs.

Ethics Statement

All our models are small, which significantly reduces the computational resources required for training and inference. All data used in the study are from open datasets. To promote transparency and reproducibility, all data and code used in this study are publicly available.

The involved university does not require IRB approval for this kind of study, which uses publicly available data without involving human participants. We do not see any other concrete risks concerning dual use of our research results. Of course, in the long run, any research results on AI methods based on large language models could potentially be used in contexts of harmful and unsafe applications of AI. But this danger is rather low in our concrete case.

References

Farrell Ackerman, James P. Blevins, and Robert Malouf. 2009. Parts and wholes: Implicative patterns in inflectional paradigms. In *Analogy in Grammar: Form and Acquisition*. Oxford University Press.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1024–1029, Denver, Colorado. Association for Computational Linguistics.

Adam Albright. 2002. Islands of reliability for regular morphology: Evidence from italian. *Language*, pages 684–709.

Adam Albright and Bruce Hayes. 2003. Rules vs. analogy in English past tenses: A computational/experimental study. *Cognition*, 90(2):119–161.

Iñaki Alegria and Izaskun Etxeberria. 2016. EHU at the SIGMORPHON 2016 shared task. a simple proposal: Grapheme-to-phoneme for inflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 27–30, Berlin, Germany. Association for Computational Linguistics.

R Harald Baayen, Yu-Ying Chuang, Elnaz Shafaei-Bajestan, and James P Blevins. 2019. The discriminative lexicon: A unified computational model for the lexicon and lexical processing in comprehension and production grounded not in (de) composition but in linear discriminative learning. *Complexity*, 2019.

Dinah Baer-Henney and Ruben van de Vijver. 2012. On the role of substance, locality, and amount of exposure in the acquisition of morphophonemic alternations. *Laboratory Phonology*, 3(2):221–249.

James P. Blevins. 2006. Word-based morphology. *Journal of Linguistics*, 42(3):531–573.

James P Blevins. 2016. *Word and paradigm morphology*. Oxford University Press.

James P. Blevins, Petar Milin, and Michael Ramscar. 2017. The Zipfian paradigm cell filling problem. In *Perspectives on Morphological Organization*, pages 139–158. BRILL.

Gilles Boyé and Gauvain Schalchli. 2019. Realistic data and paradigms: the paradigm cell finding problem. *Morphology*, 29(2):199–248.

Joan Bybee. 1995. Regular morphology and the lexicon. *Language and Cognitive Processes*, 10(5):425–455.

Joan Bybee. 2003. *Phonology and language use*, volume 94. Cambridge University Press.

Chiara Cappellaro, Nina Dumrukcic, Isabella Fritz, Francesca Franzon, and Martin Maiden. 2024. The cognitive reality of morphomes. evidence from italian. *Morphology*, 34(1):33–71.

Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1693–1703, Berlin, Germany. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Arya D. McCarthy, Katharina Kann, Sabrina J. Mielke, Garrett Nicolai, Miikka Silfverberg, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. The CoNLL–SIGMORPHON 2018 shared task: Universal morphological reinflection. In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection*, pages 1–27, Brussels. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017. CoNLL-SIGMORPHON 2017 shared task: Universal morphological reinflection in 52 languages. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 1–30, Vancouver. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared Task—Morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22, Berlin, Germany. Association for Computational Linguistics.

Verna Dankers, Anna Langedijk, Kate McCurdy, Adina Williams, and Dieuwke Hupkes. 2021. Generalising to German plural noun classes, from the perspective of a recurrent neural network. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 94–108, Online. Association for Computational Linguistics.

Fermın Moscoso del Prado Martın, Aleksandar Kostić, and R Harald Baayen. 2004. Putting the bits together: An information theoretical perspective on morphological processing. *Cognition*, 94(1):1–18.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*.

Nick C. Ellis and Richard Schmidt. 1998. Rules or associations in the acquisition of morphology? the frequency by regularity interaction in human and pdp

learning of morphosyntax. *Language and Cognitive Processes*, 13(2-3):307–336.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 634–643, San Diego, California. Association for Computational Linguistics.

Omer Goldman, David Guriel, and Reut Tsarfaty. 2022. (un)solving morphological inflection: Lemma overlap artificially inflates models' performance. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 864–870. Association for Computational Linguistics.

David Guriel, Omer Goldman, and Reut Tsarfaty. 2023. Morphological inflection with phonological features. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 613–622, Toronto, Canada. Association for Computational Linguistics.

Maria Heitmeier, Yu-Ying Chuang, and R. Harald Baayen. 2021. Modeling morphology with linear discriminative learning: Considerations and design choices. *Frontiers in Psychology*, 12.

Maria Heitmeier, Valeria Schmidt, Hendrik P. A. Lensch, and R. Harald Baayen. 2024. Is deeper always better? replacing linear mappings with deep learning networks in the Discriminative Lexicon Model. *Preprint*, arXiv:2410.04259.

Borja Herce. 2020. *A typological approach to the morphome*. Ph.D. thesis, University of Surrey.

Borja Herce and Marc Allassonnière-Tang. 2024. The meaning of morphomes: distributional semantics of spanish stem alternations. *Linguistics Vanguard*.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella Sinclair, et al. 2023. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5(10):1161–1174.

Cheonkam Jeong, Dominic Schmitz, Akhilesh Kakolu Ramarao, Anna Stein, and Kevin Tang. 2023. Linear discriminative learning: a competitive non-neural baseline for morphological inflection. In *Proceedings of the 20th SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 138–150, Toronto, Canada. Association for Computational Linguistics.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017. Neural multi-source morphological reinflection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 514–524. Association for Computational Linguistics.

Katharina Kann and Hinrich Schütze. 2016. Single-model encoder-decoder with explicit morphological representation for reinflection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 555–560, Berlin, Germany. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Jordan Kodner and Salam Khalifa. 2022. SIGMORPHON–UniMorph 2022 shared task 0: Modeling inflection in language acquisition. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 157–175, Seattle, Washington. Association for Computational Linguistics.

Jordan Kodner, Sarah Payne, Salam Khalifa, and Zoey Liu. 2023. Morphological inflection: A reality check. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6082–6101, Toronto, Canada. Association for Computational Linguistics.

Russell V. Lenth. 2024. *emmeans: Estimated Marginal Means, aka Least-Squares Means*. R package version 1.10.0.

Wang Ling, Isabel Trancoso, Chris Dyer, and Alan W Black. 2015. Character-based neural machine translation. *arXiv preprint arXiv:1511.04586*.

Ling Liu and Mans Hulden. 2020. Leveraging principal parts for morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, SIGMORPHON 2020, Online, July 10, 2020*, pages 153–161. Association for Computational Linguistics.

Ling Liu and Lingshuang Jack Mao. 2016. Morphological reinflection with conditional random fields and unsupervised features. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 36–40, Berlin, Germany. Association for Computational Linguistics.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Martin Maiden. 2011. Allomorphy, autonomous morphology and phonological conditioning in the history of the Daco-Romance present and subjunctive. *Transactions of the Philological Society*, 109(1):59–91.

Martin Maiden. 2018. *The Romance verb: Morphomic structure and diachrony*. Oxford University Press.

Martin Maiden. 2021. The morphome. *Annual Review of Linguistics*, 7:89–108.

Peter Makarov and Simon Clematide. 2018. Imitation learning for neural morphological string transduction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2877–2882, Brussels, Belgium. Association for Computational Linguistics.

Robert Malouf. 2016. Generating morphological paradigms with a recurrent neural network. *San Diego Linguistics Papers*, 6:122–129.

Andrew Nevins, Cilene Rodrigues, and Kevin Tang. 2015. The rise and fall of the L-shaped morphome: diachronic and experimental studies. *Probus: International Journal of Latin and Romance Linguistics*, 27(1):101–155.

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 922–931, Denver, Colorado. Association for Computational Linguistics.

Garrett Nicolai, Bradley Hauer, Adam St Arnaud, and Grzegorz Kondrak. 2016. Morphological reinflection via discriminative string transduction. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 31–35, Berlin, Germany. 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 the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Janet Pierrehumbert. 2001. Stochastic phonology. *Glot international*, 5(6):195–207.

David E. Rumelhart, James L. MollClelland, and the PDP Research Group. 1986. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, volume 1. The MIT Press.

Andreas Shcherbakov and Ekaterina Vylomova. 2023. Does topological ordering of morphological segments reduce morphological modeling complexity? a preliminary study on 13 languages. In *Proceedings of the 5th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, pages 120–125, Dubrovnik, Croatia. Association for Computational Linguistics.

Miikka Silfverberg and Mans Hulden. 2018a. An encoder-decoder approach to the paradigm cell filling problem. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2883–2889. Association for Computational Linguistics.

Miikka Silfverberg and Mans Hulden. 2018b. An encoder-decoder approach to the paradigm cell filling problem. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2883–2889, Brussels, Belgium. Association for Computational Linguistics.

Andrew Spencer and Mark Aronoff. 1994. Morphology by itself: Stems and inflectional classes. *Language*, 70:811.

John Sylak-Glassman. 2016. The composition and use of the universal morphological feature schema (unimorph schema). Technical report, Center for Language and Speech Processing, Johns Hopkins.

Dima Taji, Ramy Eskander, Nizar Habash, and Owen Rambow. 2016. The Columbia University - New York University Abu Dhabi SIGMORPHON 2016 morphological reinflection shared task submission. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 71–75, Berlin, Germany. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J. Mielke, Shijie Wu, Edoardo Maria Ponti, Rowan Hall Maudslay, Ran Zmigrod, Josef Valvoda, Svetlana Toldova, Francis Tyers, Elena Klyachko, Ilya Yegorov, Natalia Krizhanovsky, Paula Czarnowska, Irene Nikkarinen, Andrew Krizhanovsky, Tiago Pimentel, Lucas Torroba Hennigen, Christo Kirov, Garrett Nicolai, Adina Williams, Antonios Anastasopoulos, Hilaria Cruz, Eleanor Chodroff, Ryan Cotterell, Miikka Silfverberg, and Mans Hulden. 2020. SIGMORPHON 2020 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–39, Online. Association for Computational Linguistics.

Yiwei Wang, Yujun Cai, Muhao Chen, Yuxuan Liang, and Bryan Hooi. 2023. Primacy effect of ChatGPT. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 108–115, Singapore. Association for Computational Linguistics.

Leonie Weissweiler, Valentin Hofmann, Anjali Kantharuban, Anna Cai, Ritam Dutt, Amey Hengle, Anubha Kabra, Atharva Kulkarni, Abhishek Vijayakumar, Haofei Yu, Hinrich Schuetze, Kemal

Oflazer, and David Mortensen. 2023. Counting the bugs in ChatGPT's wugs: A multilingual investigation into the morphological capabilities of a large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6508–6524, Singapore. Association for Computational Linguistics.

Adam Wiemerslage, Kyle Gorman, and Katharina von der Wense. 2024. Quantifying the hyperparameter sensitivity of neural networks for character-level sequence-to-sequence tasks. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 674–689, St. Julian's, Malta. Association for Computational Linguistics.

Shijie Wu and Ryan Cotterell. 2019. Exact hard monotonic attention for character-level transduction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1530–1537, Florence, Italy. Association for Computational Linguistics.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the transformer to character-level transduction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1901–1907, Online. Association for Computational Linguistics.

A Appendices

A.1 Accuracies

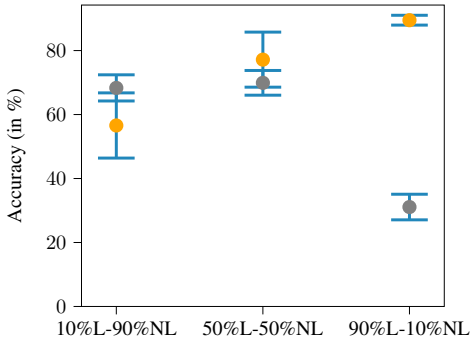

Figure 4: Mean and confidence intervals of stem sequence accuracies for L-shaped and NL-shaped verbs across frequency conditions. Gray: NL-shaped, Orange: L-shaped.

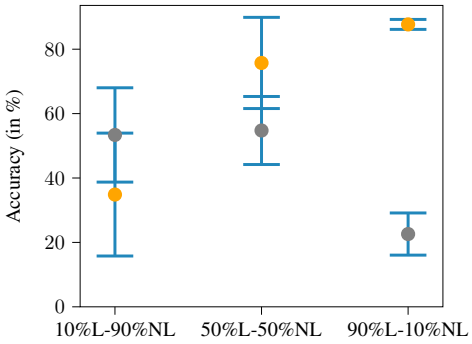

Figure 5: Mean accuracies and confidence intervals of overall sequence accuracies for L-shaped and NL-shaped verbs across frequency conditions, evaluated using the model with a batch size of 512. Gray: NL-shaped, Orange: L-shaped.

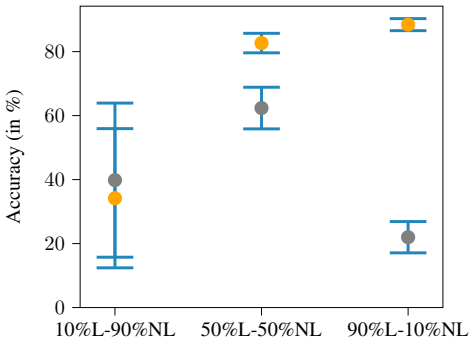

Figure 6: Mean accuracies and confidence intervals of overall sequence accuracies for L-shaped and NL-shaped verbs across frequency conditions, evaluated using the model with a batch size of 800. Gray: NL-shaped, Orange: L-shaped.

A.2 Cell combinations

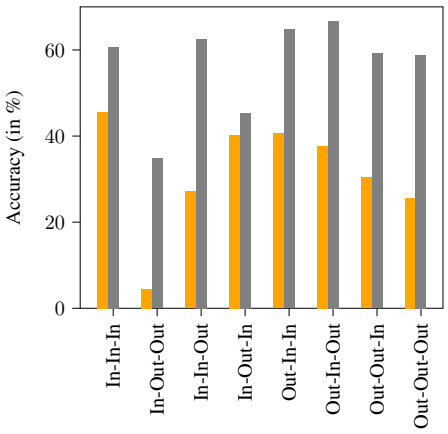

Figure 7: Cell combination accuracies for 10L-90NL condition. Gray: NL-shaped, Orange: L-shaped.

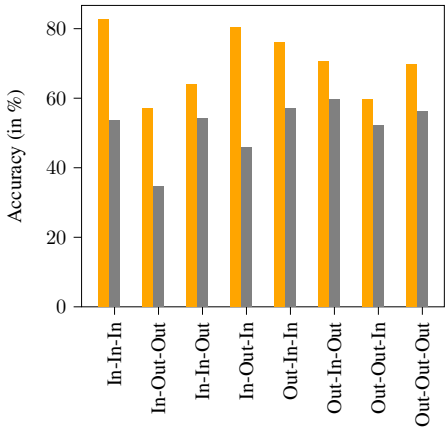

Figure 8: Cell combination accuracies for 50L-50NL condition. Gray: NL-shaped, Orange: L-shaped.

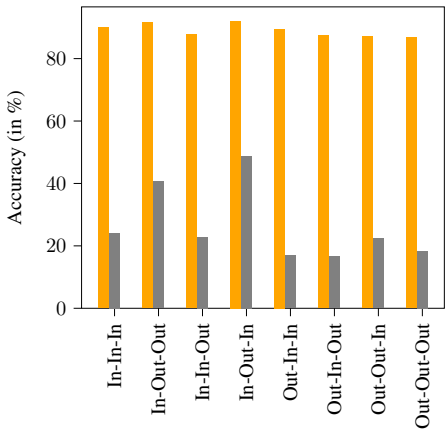

Figure 9: Cell combination accuracies for 90L-10NL condition. Gray: NL-shaped, Orange: L-shaped.

A.3 Memorization and Generalization task

Knowledge state	Condition	Prob.	SE	asymp.LCL	asymp.UCL
Generalization	10%L-90%NL	0.216	0.0150	0.188	0.247
Memorization	10%L-90%NL	0.252	0.0183	0.218	0.290
Generalization	50%L-50%NL	0.740	0.0260	0.686	0.788
Memorization	50%L-50%NL	0.627	0.0351	0.556	0.693
Generalization	90%L-10%NL	0.450	0.0353	0.382	0.520
Memorization	90%L-10%NL	0.754	0.0301	0.690	0.808

Table 3: Estimated marginal means for knowledge state and frequency condition combinations

Memorization					
Contrast	odds.ratio	SE	null	z.ratio	p.value
10%L-90%NL / 50%L-50%NL	0.2009	0.02336	Inf	-13.806	<.0001
10%L-90%NL / 90%L-10%NL	0.1100	0.01394	Inf	-17.420	<.0001
50%L-50%NL / 90%L-10%NL	0.5476	0.08795	Inf	-3.749	0.0005
Generalization					
Contrast	odds.ratio	SE	null	p.value	
10%L-90%NL / 50%L-50%NL	0.0969	0.00949	Inf	-23.833	<.0001
10%L-90%NL / 90%L-10%NL	0.3369	0.03802	Inf	-9.642	<.0001
50%L-50%NL / 90%L-10%NL	3.4750	0.47015	Inf	9.207	<.0001

Table 4: Pairwise comparisons of frequency condition levels for each knowledge state

A.4 Consonant-pair analysis

Stem-final consonants	No. of occurrences
[s]-[sk]	141
[n]-[ng]	53
[ç]-[x]	25
[s]-[g]	15
[]-[jg]	14
[rç]-[rx]	10
[nç]-[nx]	10
[l]-[lg]	4
[lç]-[lx]	4
[s]-[sg]	2
[b]-[p]	1

Table 5: Number of alternating stem final-consonant pairs of all the L-shaped verbs.

	Consonant-pairs	Freq. in Test	Freq. in Train
Run 1	[s]-[sk]	3	8
	[n]-[ng]	2	4
	[nç]-[nx]	1	1
	[lç]-[lx]	1	0
Run 2	[s]-[sk]	3	13
	[n]-[ng]	1	2
	[nç]-[nx]	1	1
	[s]-[g]	1	0
Run 3	[s]-[sk]	4	10
	[s]-[g]	2	1

Table 6: Frequency of stem-final consonants in train and test in 10%L-90%NL condition.

	Consonant-pairs	Freq. in Test	Freq. in Train
Run 1	[s]-[sk]	19	20
	[n]-[ng]	3	10
	[nç]-[nx]	2	0
	[s]-[g]	3	2
	[l]-[lg]	1	0
	[ ]-[jg]	3	0
	[lç]-[lx]	1	1
Run 2	[s]-[sk]	22	15
	[ç]-[x]	2	5
	[n]-[ng]	3	7
	[nç]-[nx]	1	1
	[lç]-[lx]	1	1
	[s]-[g]	2	3
	[s]-[sg]	1	0
Run 3	[s]-[sk]	14	23
	[s]-[g]	4	0
	[n]-[ng]	2	12
	[ ]-[jg]	3	4
	[l]-[lg]	1	1
	[rç]-[rx]	1	2
	[nç]-[nx]	2	1
	[ç]-[x]	2	4

Table 7: Frequency of stem-final consonants in train and test in 50%L-50%NL condition.

	Consonant-pairs	Freq. in Test	Freq. in Train
Run 1	[s]-[sk]	32	97
	[ç]-[x]	5	18
	[ ]-[jg]	5	5
	[nç]-[nx]	2	7
	[rç]-[rx]	3	6
	[n]-[ng]	6	40
	[lç]-[lx]	1	2
	[s]-[g]	2	11
Run 2	[s]-[sk]	22	104
	[ç]-[x]	2	5
	[rç]-[rx]	1	7
	[s]-[g]	4	10
	[l]-[lg]	2	2
	[ ]-[jg]	4	8
	[nç]-[nx]	3	7
	[s]-[sg]	1	1
Run 3	[s]-[sk]	25	103
	[nç]-[nx]	4	6
	[s]-[g]	2	11
	[ç]-[x]	7	18
	[lç]-[lx]	2	1
	[n]-[ng]	10	35
	[l]-[lg]	2	2
	[rç]-[rx]	1	6
	[ ]-[jg]	1	9

Table 8: Frequency of stem-final consonants in train and test in 90%L-10%NL condition.

A.5 Confusion Matrix

10%L-90%NL condition										
	P [s]-[s]	P [s]-[sk]	P [s]-[g]	P [n]-[ng]	P [s]-[]	P [s]-[d]	P [s]-[sg]	P [n]-[n]	P [n]-[rb]	Accuracy
G [s]-[sk]	303	**857**	0	0	48	41	0	0	0	68.61%
G [s]-[g]	235	21	**99**	0	8	0	15	0	0	26.19%
G [n]-[ng]	0	0	0	**281**	0	0	0	56	26	77.41%
50%L-50%NL condition										
	P [s]-[sk]	P [s]-[g]	P [n]-[ng]	P [s]-[s]	P [s]-[d]	P [s]-[ʃ]	P [s]-[jg]	P [n]-[n]	P [n]-[mp]	Accuracy
G [s]-[sk]	**6170**	32	0	523	73	70	0	0	0	89.84%
G [s]-[g]	112	**613**	0	210	0	0	76	0	0	60.63%
G [n]-[ng]	0	0	**853**	0	0	0	0	139	29	83.55%
90%L-10%NL condition										
	P [s]-[sk]	P [n]-[ng]	P [ç]-[x]	P [s]-[s]	P [s]-[rs]	P [n]-[n]	P [n]-[sk]	P [n]-[n]	P [ç]-[ç]	Accuracy
G [s]-[sk]	**9473**	0	0	570	36	70	0	0	0	93.34%
G [n]-[ng]	0	**3394**	0	0	0	130	0	139	0	92.66%
G [ç]-[x]	0	0	**1954**	0	0	0	0	0	183	91.44%

Table 9: Confusion matrix for the top 5 most erroneous consonant-pairs (considering mean values) for L-shaped verbs across conditions. G = Gold, P = Prediction. Bold indicates the correct predictions.