The Effect of Data Partitioning Strategy on Model Generalizability: A Case Study of Morphological Segmentation

Zoey Liu
Department of Linguistics
University of Florida
liu.ying@ufl.edu

Bonnie J. Dorr
Computer & Information Science & Engineering
Florida Institute for National Security
University of Florida
bonniejdorr@ufl.edu

Abstract

Recent work to enhance data partitioning strategies for more realistic model evaluation face challenges in providing a clear optimal choice. This study addresses these challenges, focusing on morphological segmentation and synthesizing limitations related to language diversity, adoption of multiple datasets and splits, and detailed model comparisons. Our study leverages data from 19 languages, including ten indigenous or endangered languages across 10 language families with diverse morphological systems (polysynthetic, fusional, and agglutinative) and different degrees of data availability. We conduct large-scale experimentation with varying sized combinations of training and evaluation sets as well as new test data. Our results show that, when faced with new test data: (1) models trained from random splits are able to achieve higher numerical scores; (2) model rankings derived from random splits tend to generalize more consistently.

1 Introduction

Evaluations of computational models in natural language processing typically rely on a single dataset, though there are exceptions, such as high-resource languages like English, which have multiple datasets available for specific tasks (Rajpurkar et al., 2016; Kwiatkowski et al., 2019; Warstadt et al., 2020). Such datasets are typically provided with *one* partitioning, including at least a training and test set, with an optional validation set depending on data availability (Gauthier et al., 2016; Cotterell et al., 2015). The single data split is typically determined by shared task organizers (Kurimo et al., 2006), benchmark designers (Wang et al., 2018), and ground-breaking paper authors (Collins, 2002). The rationale behind the particular data partition, however, is not always clear or mentioned at all (cf. de Marneffe et al. (2021)).

Recent work has called into question the adoption of one data split (Gorman and Bedrick, 2019; Liu et al., 2023; Kodner et al., 2023) or one dataset (Søgaard et al., 2021) for model evaluation. These previous studies point out that individual model performance as well as model rankings derived from just one single split of training-(validation)-test sets *may fail to generalize* when applied to an alternative split of the same dataset, or even to new unseen data from the same domain. Thus, drawing conclusions based on a single data partition has the potential for being unreliable.

This study investigates the effect of different data partitioning methods on model generalizability in cross-linguistic scenarios with *varying data availability*. We use morphological segmentation as the testbed, i.e., the task of decomposing a word into its component morphemes (*avocados* → *avocado* + *s*). While a range of studies have been undertaken to explore the impact of data partitioning strategies on generalizability of model performance (see Section 2 for some examples), it is safe to say that no consensus has been reached regarding the specific choices of data partition strategies. This is largely due to the fact that these studies face several important limitations, which we describe below.

Lack of language diversity First, a considerable portion of prior work (Gorman and Bedrick, 2019; Søgaard et al., 2021) predominantly focuses on English (cf. Bender (2019); Ducel et al. (2022)). It is possible, however, that an optimal data partitioning strategy, if one exists, is dependent on the languages (and tasks) under investigation. This is because the typological traits of the languages can have an impact on the distributions of the resulting training/test sets and new unseen data. For example, if languages exhibit greater morphological regularity, alternative data partitioning approaches might yield comparable model performance.

Lack of multiple datasets and data splits Building on the first point, the tasks investigated in previous literature often enjoy ample data availability. For high-resource languages with abundant data

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 2851–2864

for a given task, it is often assumed (implicitly or explicitly) that the selected dataset or data split adequately represents the task or language. Therefore, a data partitioning strategy that fares well on the same dataset or data split is expected to yield models that generalize reasonably to new unseen data, particularly from the same domain. However, in scenarios with constraints on data availability, the representativeness of the chosen dataset or data split becomes questionable. What kind of data partitioning strategy is appropriate to apply when facing different extents of data availability thus remains an open question. Liu et al. (2023) address the aforementioned issues to some extent, but lack evaluation of trained models on new test samples. Although Søgaard et al. recommend inclusion of multiple test sets, they do not consistently experiment accordingly for each task.

Lack of model comparisons Finally, while Gorman and Bedrick compare a number of POS taggers, Søgaard et al. and Liu et al. apply one model for each task. Thus, they fail to provide a detailed analysis of model rankings and how these rankings may be affected by different partitioning strategies. It remains unclear if these rankings would still hold when considering new test samples.

Taking a data-driven approach, this work transcends that of prior approaches in several respects:

- We attend to a typologically diverse set of 19 languages from ten language families, covering polysynthetic, fusional, and agglutinative morphological systems. These languages have different amounts of data available pertaining to morphological segmentation. In addition, ten of these languages are indigenous or endangered languages, painting a typologically rich set of language samples for our study.
- We compare four model architectures in order to analyze model rankings.
- Perhaps most importantly, we conduct a sequence of large-scale experiments, varying both the combinations of training and evaluation sets along with their respective sizes. To evaluate the generalizations of model performance resulting from different data partitioning strategies, we generate new test samples of different sizes as well.

In what follows, Section 2 describes recent studies to explore the respective effect of different data splits on model performance. Section 3 presents our experimentation, including dataset creation and evaluation of four model architectures to probe model generalizability. Section 4 provides an analysis, answering questions about the impact of data partitioning strategies on model generalizability. Section 5 concludes with possible avenues for future work. Finally, we address the limitations of our approach, followed by a statement on ethics and broader impact.

2 Related Work

Data partitioning strategies Recent research proposes different strategies to address the question of data split impact on model generalizability (see Table 1 for a summary of comparisons between previous work and our studies.).

Study	Multilingual	Including Resource-constrained scenarios	Multi-datasets	Multi-models
G&B	✗	✗	✗	✓
SEBF	✗	✗	not always	✗
LSP	✓	✓	✗	✗
Ours	✓	✓	✓	✓

Table 1: Comparisons of experimental setups between G&B (Gorman and Bedrick, 2019), SEBF (Søgaard et al., 2021), LSP (Liu et al., 2023), and our study here.

Gorman and Bedrick (2019) conduct a series of replication and reproduction experiments on part-of-speech (POS) tagging using the Wall Street Journal (WSJ) from the Penn Treebank (Marcus et al., 1993). Their work re-evaluates the performance of eight POS taggers previously claimed to achieve state-of-the-art performance on one split of the WSJ dataset from Collins (2002). They refer to this as "standard split", dividing the WSJ dataset as follows: 00–18 as the training set, 19-21 as the development set, and 22-24 for test. They then compare the ranking of these taggers on the pre-defined split with their rankings on multiple randomly generated splits of the same dataset. The study reveals noticeable inconsistencies in model rankings between the standard split and random splits. As a result, the authors recommend adopting random splits for when comparing the performance of different model architectures.

Søgaard et al. (2021) counterargue the proposal by Gorman and Bedrick (2019). With six tasks in English ranging from POS tagging to news classification, Søgaard et al. illustrate that random splits over-estimate individual model performance when it comes to new in-domain data (new test samples). By contrast, more reliable numerical estimates are obtained by adversarial splits, which partition a

dataset to ensure the test set distribution is as different as possible from that of the training set.

In a study that compares various data partitioning strategies for automatic speech recognition evaluation, Liu et al. (2023) show that random splits, rather than adversarial splits, offer a more comprehensive capability assessment for a given acoustic model architecture. This finding is particularly relevant when considering five indigenous endangered languages with minimal training resources.

Collectively, it is not clear, based on existing findings, which data split strategies are more capable of yielding models with more generalizable performance. We consider that the lack of consensus among prior studies is largely due to the lack of thorough experimentation pertaining to the number (and types) of languages, datasets and splits, as well as model architectures employed. This study tackles these limitations by providing a sequence of data-driven experiments, with the goal of providing *empirical* evidence for the capabilities of different data partitioning strategies.

Morphological segmentation Morphological segmentation has received considerable interest in the literature. Previous studies have demonstrated that incorporating morphological information effectively eliminates data sparsity issues for a variety of downstream NLP tasks. These tasks include but are not limited to automatic speech recognition for languages such as Vietnamese (Le and Besacier, 2009) and Finnish and Turkish (Kurimo et al., 2006), as well as machine translation for various language pairs (e.g., English $\rightarrow$ Finnish (Clifton and Sarkar, 2011); Raramuri/Shipibo-Konibo $\leftrightarrow$ Spanish (Mager et al., 2022)).

3 Experiments

This section introduces our experimentation to investigate the impact of different data partitioning strategies on model generalizability for morphological segmentation. We first present the details regarding the data used in our experiments, including the languages/families contained therein. We then explore the dataset construction process and describe the model architectures applied.

3.1 Data sources

We adopt morphological segmentation data for a total of 19 languages, spanning 10 language families, to join our experiments. Table 2 provides relevant descriptive information and the prior works that synthesize the morphological segmentation data for the languages. Below we introduce the original data sources for each language (Bender and Friedman, 2018). Among these languages, eight are polysynthetic indigenous/endangered Mexican languages (Mexicanero, Nahuatl, Yorem Nokki, Wixarika, Raramuri, Popoluca, Tepehua), which are all from the Yuto-Aztecan language family, and Shipibo-Konibo, which is from the Panoan language family primarily spoken in Peru and Brazil. The Raramuri data originally come from work by Caballero (2010) and a dissertation (Caballero, 2008). The Shipibo-Konibo data are (largely) taken from a dependency treebank (Vasquez et al., 2018). Morphological segmentation data for Mexican languages are digitized from the Archive of Indigenous Language.

Seneca and Hupa are critically endangered Native American languages from the Iroquoian and Dene/Athabaskan language family respectively; the former is primarily spoken in New York State and Ontario, while the latter is the ancestral language of the Hoopa Valley Tribe in Northern California. Seneca data are digitized from a grammar book (Bardeau, 2007), while Hupa data consist of examples from several archival collections (Curtin, 1888-1889; Kroeber, 1900-1906; Woodward, 1953), along with words taken from ongoing fieldwork with an elder from the Hupa speech community. Both languages have polysynthetic morphological properties.

Next, we have two fusional, Indo-European languages: (1) English data come from the Morpho Challenge shared task for unsupervised approaches to morphological segmentation (Kurimo et al., 2010); (2) German data are harnessed from the CELEX lexical database (Baayen et al., 1996).

The remaining seven languages are agglutinative. Finish and Turkish data come from the Morpho Challenge. Indonesian data are from an Indonesian-English bilingual corpus.[1] Zulu data are collected from the Ukwabelana Corpus (Spiegler et al., 2010). Lastly, the morphological segmentation data for Akan, Swahili, and Tegulu come from efforts in the DARPA Low Resource Languages for Emerging Incidents (LORELEI) Program (Mott et al., 2020).

3.2 Data partitioning strategy

We explore two different data partitioning strategies: **random** and **adversarial**. Random splits di-

[1] https://github.com/desmond86/Indonesian-English-Bilingual-Corpus

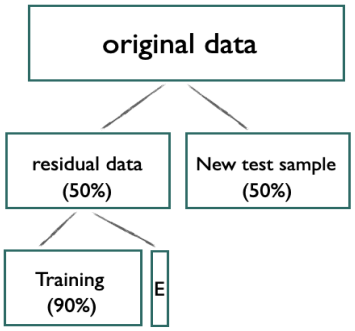

Figure 1: Simple illustration of a single dataset construction process for our experiments: given the original dataset of a language, a new test sample is constructed either randomly or adversarially (in this case, the new test sample accounts for 50% of the original dataset); the residual data from the original data is then divided into a training and an evaluation set (*E*) at a fixed 9:1 ratio, via random or adversarial splits.

vide a dataset into training and test data randomly, whereas adversarial splits partition the dataset such that the Wasserstein distance (Arjovsky et al., 2017; Søgaard et al., 2021) between the morpheme distributions of the resulting training and test data is maximized. The aim of employing adversarial splits is to create test data that are as distant or *different* as possible from the training data. Thus, model training on adversarial splits may pose greater challenges compared to random splits.

3.3 Dataset construction

Dataset construction proceeds as follows (see also Table 2 and Figure 1). From the initial data of a given language, we first select all the unique words. The main motivation for this choice is that in practice, if a word in the test data is already included in the training data, then its morphological segmentation annotations can be directly copied from its annotations in the training data. We refer to the resulting dataset that includes all the unique words as the *original dataset* in our experiments.

We determine a range of *new test sample* sizes: {10%, 20%, 30%, 40%, 50%} of the original dataset. Given each size, we *randomly* partition the original dataset into a new test sample and a *residual dataset*, 10 times. In other words, for each size, we construct 10 new test samples. We consider the new test samples as approximations of new unseen data that is aside from what would normally be included in a typical dataset for practices of model training and evaluation. Ideally, one would use perhaps data of different domains to be new unseen data. That said, due to data availability limitation, we are not able to find datasets covering separate domains for the languages studied here.

For each new test sample, we use the corresponding residual dataset to build training and evaluation (eval) sets via each of the two data partitioning strategies of interest: random and adversarial splits.[2] Given each data partitioning strategy, we split the residual dataset three times, aiming for a 9:1 ratio between the resulting training and eval sets each time. As such, each individual new test sample is paired with 3 training-eval sets from random splits and another three from adversarial splits. This enables a direct comparison of models trained using the two data partitioning strategies, allowing us to determine which strategy leads to models with better performance on new test samples.

Moreover, this approach results in a total of 30 combinations consisting of a training set, an eval set, and a new test sample for each combination of a new test sample size and data partitioning strategy. Doing so motivates us to explore the variability across different datasets of the same size and ensure the generalization of our observations.

Lastly, we repeat the full process described above, except that this time the new test samples are generated *adversarially* (i.e., partitioning the original dataset into new test samples and their corresponding residual dataset via adversarial splits).

We note that our focus on data partitioning strategies pertains to *how the residual dataset is divided*; the reason we employ different ways of generating new test samples (randomly and adversarially) is solely to see if observations will hold qualitatively, regardless of how the new test samples are derived.

3.4 Model architectures

We employ four model alternatives from two broad model classes: conditional random field (CRF) (Lafferty et al., 2001), and neural sequence-to-sequence (seq2seq) models. CRF models are log-linear discriminative models that treat morphological segmentation as a sequence tagging task. We experiment with first-order CRF. Given a word w, we add a start (<w>) and an end (</w>) symbol. For each character w_i in the word, where i repre-

[2]We also explored **heuristic** splits as a third strategy in initial experiments. These splits are determined by considering the average morpheme count and length. In our automated search for a metric threshold in the residual data, to divide it into training and eval sets, we identify words with an average number of morphemes equal to or greater than this threshold and assign them to the eval set. However, we find that for most of our experimental setups, no such threshold exists.

Language	Language family	Morphological system	# word type	Ave. morph len	Data available by
Mexicanero	Yuto-Aztecan	Polysynthetic	882	3.93	Kann et al. 2018
Nahuatl	"	"	1,096	3.90	"
Yorem Nokki	"	"	1,050	3.61	"
Wixarika	"	"	1,350	3.26	"
Raramuri	"	"	914	3.57	Mager et al. (2022)
Popoluca	"	"	898	4.31	Mager et al. (2020)
Tepehua	"	"	816	5.37	"
Shipibo-Konibo	Panoan	"	1,096	4.15	Mager et al. (2022)
Seneca	Iroquoian	"	5,425	2.98	Liu et al. (2021)
Hupa	Dene/Athabaskan	"	595	3.99	Curtin (1888-1889) Kroeber (1900-1906); Woodward (1953); linguistic fieldwork
English	Indo-European	Fusional	1,686	4.09	Cotterell et al. (2015)
German	"	"	1,751	3.82	"
Finnish	Uralic	Agglutinative	1,835	4.03	"
Turkish	Turkic	"	1,763	3.38	"
Indonesian	Austronesian	"	3,500	4.98	"
Zulu	Niger-Congo	"	10,040	2.37	"
Akan	"	"	2,046	2.49	Mott et al. (2020)
Swahili	"	"	2,023	2.91	"
Telugu	Dravidian	"	2,007	4.07	"

Table 2: Descriptive statistics for the initial morphological segmentation data of each language in our experiments; *Data available by* refers to the prior work that makes the initial morphological segmentation data of the corresponding language(s) available (with the exceptions of Hupa).

sents the character's index position, we assign one of six labels: `START` (for <w>), `END` (for </w>), `S` (for any single-character morpheme), and `B` (beginning), `M` (middle), or `E` (end) for characters within a multi-character morpheme. As an illustration, the word *avocados* will have the following sequence of segmentation labels:

```
<w>    a v o c a d o s   </w>
START  B M M M M M E S   END
```

Lastly, for each character w_i in w, we curate a feature set from local n-gram (sub-)strings, as input to a first-order CRF model, to predict the corresponding label for w_i. All CRF models are implemented using the `python-crfsuite` framework.[3]

For seq2seq, we use `fairseq` (Ott et al., 2019) to explore three different encoder-decoders: LSTM, TRANSFORMER, and TRANSFORMER_TINY.[4] For each encoder-decoder architecture, the model input is always the word itself as a sequence of letters (with space between every two consecutive letters), and the model output contains an extra exclamation point (!) as indication of morpheme boundary.

```
INPUT    a v o c a d o s
OUTPUT   a v o c a d o ! s
```

[3] https://python-crfsuite.readthedocs.io/en/latest/

[4] https://fairseq.readthedocs.io/en/latest/models.html

All seq2seq models are implemented using the default parameters: for the LSTM-based architecture, all embeddings have 512 dimensions; both the encoder and the decoder contain one hidden layers with 512 hidden units in each layer; TRANSFORMER has 6 encoder-decoder layers, 8 self-attention heads, an embedding size of 512, and 2048 hidden units in the feed-forward layers; TRANSFORMER_TINY has 2 encoder-decoder layers, 2 self-attention heads, with the embedding dimension and feed-forward layer dimension both being 64.

In all experimental configurations conducted in this study, the parameter implementation for each model architecture remains the same for all languages (see also Appendix A). The model evaluation is performed using the $F1$ score as the metric.

4 Analysis

Our analysis seeks to address two questions:

(1) Which data partitioning strategy leads to more accurate numerical "guesses" of, as well as better, individual model performance for the new test samples?

(2) How do different data partitioning strategies affect the generalization of model rankings from the eval sets to new test samples?

Note that for each research question, we perform analysis of each individual language first, then fo-

cus on a summary of aggregated averages across languages; languages with idiosyncratic patterns, however, are noted when necessary. Throughout our analysis, we first describe results from cases where the new test samples are derived randomly. We then move onto settings where the new test samples are generated adversarially, in order to see if there are notable similarities and differences in the observations between the two.

4.1 Individual model performance

This section analyzes estimates of individual model performance using different data partitioning strategies (applied to residual data), *when the new test samples are generated randomly*. Given each language, for every model architecture, we measure the average $F1$ score difference between the eval sets and their corresponding new test samples. A higher average $F1$ score difference indicates that the performance of a given model does not generalize well from the eval sets to new test samples.

Across the 19 languages, adversarial splits lead to much larger score differences for all four models, with the scores for new test samples consistently higher than those for eval sets (Table 3). On the contrary, for random splits, there is no noticeable score difference for any of the model architectures. (Detailed language-by-language results are in Table 7 in Appendix B). Collectively, these patterns suggest when focusing solely on the achieved scores of a model, random splits provide more reliable numerical estimates compared to adversarial splits.

Model	Split	Eval sets	New test
CRF	random	0.80	0.80
	adversarial	0.76	0.79
TRM_TINY	random	0.68	0.68
	adversarial	0.59	0.67
LSTM	random	0.67	0.67
	adversarial	0.62	0.65
TRM	random	0.56	0.56
	adversarial	0.48	0.54

Table 3: Individual model performance ($F1$) for eval sets and new test samples averaged across languages, when the new test samples are generated **randomly**.

Furthermore, we compare which data partitioning strategy results in higher scores for the new test samples. As shown in Table 3, random splits consistently lead to (slightly) better model performance across the four model architectures. Among the different models, the largest performance gap from the two data partitioning strategies is observed for LSTM (0.02) and TRANSFORMER (0.02). Similar observations exist when analyzing variously sized training sets and new test samples.

We carry out the same analysis for cases *where the new test samples are derived adversarially* (Table 4). We find that adversarial splits of the residual data *instead* lead to lower $F1$ score difference (0.09) on average across settings, compared to random splits (0.15). This holds mostly when breaking down by languages and individual model architectures as well. That said, randomly partitioning the residual data yields better average model performance for new test samples. (See Table 8 in Appendix B for language-by-language results.) These results suggest that if one were to care mainly about achieving a higher numerical score on additional data with a given model architecture, random splits would be a more suitable option.

Model	Split	Eval sets	New test
CRF	random	0.80	0.65
	adversarial	0.67	0.59
TRM_TINY	random	0.69	0.52
	adversarial	0.55	0.46
LSTM	random	0.66	0.52
	adversarial	0.55	0.45
TRM	random	0.55	0.41
	adversarial	0.44	0.37

Table 4: Individual model performance ($F1$) for eval sets and new test samples averaged across languages, when the new test samples are generated **adversarially**.

4.2 Model ranking

We now turn to studying the effect of different data partitioning strategies on model ranking generalizations, *when the new test samples are derived randomly*. For each of the two data partitioning strategies, given every combination of a training set, an eval set, and a new test sample, we derive the ranking of the four model architectures based on their F1 scores (averaged across 3 random seeds) on the eval set ($Ranking\_eval$) and the new test sample ($Ranking\_new$), respectively. (Again, based on how we construct the datasets initially, the two $F1$ scores are predicted by the same model from the training set, thereby directly comparable).

Best overall model ranking We compute the best overall model ranking (e.g., the most frequent $Ranking\_eval$) for both the eval sets and the new test samples, considering each data partitioning strategy. Comparing results across languages, for both random and adversarial partitions, the best

overall ranking is CRF > TRANSFORMER_TINY > LSTM > TRANSFORMER. This ranking holds for twelve out of the 19 languages examined here (Table 5), including examples from all three morphological systems covered in this study. Additionally, for these languages, there are on average noticeable $F1$ score differences between CRF (the best model) and TRANSFORMER_TINY (the second best model) for both eval sets (random: 0.12; adversarial: 0.16) and new test samples (random: 0.12; adversarial: 0.12).

CRF > TRANSFORMER_TINY > LSTM > TRANSFORMER
Mexicanero, Nahuatl, Yorem Nokki, Raramuri, Popoluca, Shipibo-Konibo, Hupa, English, Turkish, Indonesian, Swahili, Telugu
CRF > LSTM> TRANSFORMER_TINY > TRANSFORMER
Wixarika, Tepehua, Seneca, German, Finnish, Zulu
CRF > LSTM > TRANSFORMER > TRANSFORMER_TINY
Akan

Table 5: Results for the best overall rankings, when the new test samples are generated **randomly**.

The best overall ranking above is followed by an alternative that is the best for six other languages (CRF > LSTM> TRANSFORMER_TINY > TRANSFORMER), covering different morphological properties as well (Table 5). The main difference between the two rankings pertains to LSTM and TRANSFORMER_TINY. For languages where the second best overall ranking applies, the average $F1$ difference between LSTM and TRANSFORMER_TINY is mostly smaller than 0.02. Again, there are notable average $F1$ differences between CRF and the second best performing model, LSTM, for eval sets (random: 0.12; adversarial: 0.18) and new test samples (random: 0.12; adversarial: 0.12).

Overall model ranking generalizability We also examine model ranking consistency between each eval-set/new-test pair by measuring the proportion of cases where $Ranking\_new$ is the same as $Ranking\_eval$ for training/eval/new-test combinations. Given that the $F1$ score difference between LSTM and TRANSFORMER_TINY is relatively minimal (see above and Table 3), we collapse the two best rankings described above when measuring model ranking consistency. On average, the consistency of model rankings between eval sets and the new test samples ($Ranking\_eval$ = $Ranking\_new$) is higher for random splits (91.47%) than for adversarial splits (90.26%). This pattern persists for most of the languages.

We now investigate settings with *adversarially generated new test samples*. The two best overall model rankings are the same as the cases above, where new test samples are constructed randomly. Regarding the consistency of model rankings, again, we merge the two best overall rankings. The results show that the average proportion of cases where $Ranking\_new$ and $Ranking\_eval$ are the same is higher for random splits (78.67%) in contrast to observations from adversarial splits (75.79%). The average numerical discrepancy here (2.88%) is also larger than what is reported above for randomly constructed new test samples (91.47%-90.26%=1.21%). These findings indicate random splits possibly yield more reliable model ranking results in the face of new test samples.

4.3 Variation across datasets

Thus far, our analysis focuses on scores averaged across datasets. Recall that, for each language and each combination of new test sample size and data partitioning strategy, we construct 30 sets comprised of a training set, an eval set, and a new test sample. This section aims to better understand the extent of variability in model performance between the two data partitioning strategies across new test samples of the same sizes. This can, in turn, shed light on the reliability of our prior analysis, which depends on average scores across different settings.

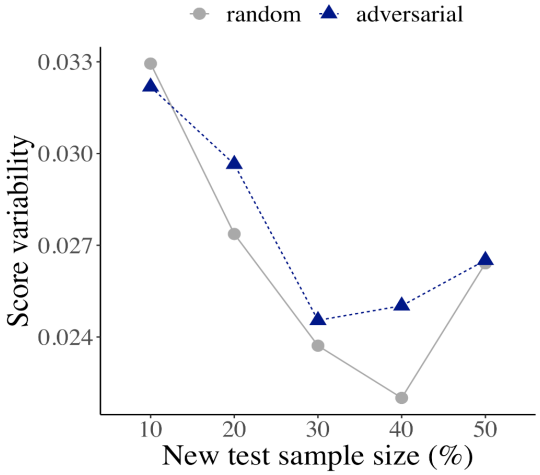

Figure 2: Score variability for every eval set size (%) given each data partitioning strategy averaged across languages and model architectures, when the new test samples are generated **randomly**.

For each new test sample size and data partitioning strategy (for a given language), we calculate $F1$ variability (standard deviation) for each model architecture across the 30 combinations of a train-

ing set, an eval set, and a new test sample. We then measure the average score variability across languages and model architectures. As shown in Figure 2, when the new test samples are constructed via random sampling, the average score variability exhibits comparable values for almost all new test sample sizes between random and adversarial splits. The score variability values predominantly fall below 0.03; this suggests that there is a small amount of variation across datasets with the same new test sample sizes. Compared to the other three neural alternatives, CRF exhibits the least performance variation; across model architectures, TRANSFORMER_TINY demonstrates the highest $F1$ variability for both data partitioning strategies.

Similar observations are found for cases where the new test samples are constructed adversarially. These patterns further validate our previous findings on individual model performance and model rankings averaged across datasets.

4.4 Regression analysis

Having established that random splits lead to comparatively better individual model performance and more generalizable model rankings on new test samples, this section aims to add statistical rigor to our findings. To achieve this, we resort to regression analysis. Our variable of interest is the data partitioning strategy applied to the residual data, which has interaction terms with various control variables: the method of deriving new test samples (randomly or adversarially), morpheme overlap (the proportion of morphemes in the eval set that occur in the training set), and the relative ratios between training and eval sets for the average number of *morphemes* per word and the average number of morpheme *types* per word. Lastly, we control for the model architecture applied.

Ideally, we would fit one mixed-effect regression model including all factors described above, with the language as a random effect. The full dataset resulting from all experiments, however, is quite large (N = 91,200); therefore we turn to fitting one linear regression model for each language instead. The goal here is to determine whether the superior performance of random split *is an observation that can be found for all, or most of the languages, regardless of their respective dataset size*.

The regression coefficients for the data partitioning strategy are presented in Table 6 (see Table 9 in Appendix C for coefficients of other control variables). A significantly positive coefficient value means random splits lead to significantly higher $F1$ scores, in contrast to adversarial splits. This pattern is evident in 16 out of nineteen languages, with the exceptions of Wixarika and German, where random splits appear to have a significantly negative effect on model performance, and Akan, where there is no pronounced difference between the two data partitioning strategies. The statistics overall further corroborate our prior observations.

Language	**Data partitiong strategy** β
Mexicanero	0.05***
Nahuatl	0.34***
Yorem Nokki	0.67***
Wixarika	-0.30***
Raramuri	0.42***
Popoluca	0.31***
Tepehua	0.08**
Shipibo-Konibo	0.80***
Seneca	1.12***
Hupa	0.14***
English	0.33***
German	-0.19***
Finnish	0.50***
Turkish	0.46***
Indonesian	0.29***
Zulu	1.90***
Akan	0.26
Swahili	0.20*
Telugu	0.10**

Table 6: Regression coefficients (β) of data partitioning strategy for each language; a positive coefficient value indicates that randomly splitting the residual data has a positive effect on $F1$ scores, while a negative value denotes the opposite; the number of * suggests significance level: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$.

5 Conclusions

This study investigates the impact of data partitioning strategies on model generalizability, using morphological segmentation as a test case, drawing data from 19 typologically diverse languages, including ten indigenous/endangered languages. Our results demonstrate that, independent of the morphological properties of the languages, random splits, in contrast to adversarial splits, yield: (1) better model performance, and (2) more reliable model rankings on new test data. These patterns hold across varying sized combinations of training and eval sets, as well as new test samples, as evidenced by the minimal variation in model performance across datasets for the languages examined. The findings are also supported by our statistical regression analysis, where random splits are shown to have a pronounced positive impact on model performance; this pattern holds for most languages,

in spite of the fact that each individual language has a different dataset size.

It is worth noting that the average $F1$ score differences between random and adversarial splits on the new test samples are much larger across model architectures, when the new test samples are generated adversarially, in comparison to when they are derived randomly (Section 4.1). What's more, the trend that random splits yield more consistent model rankings is stronger when facing adversarially constructed new test samples as well (Section 4.2). Recall that adversarial samples are posed to be as distant from the training data as possible. These tendencies suggest that when facing more challenging new test data in the wild (challenging *relative to the training data*), there is potentially more benefit in applying random splits, at least in the case of morphological segmentation.

While random splits seem to outperform adversarial splits in our study, we do not wish to draw the same conclusions for other tasks. With the methodologies outlined here, for future work, we would like to expand to different tasks from a cross-linguistic angle. In particular, we are interested in settings where the languages have *a spectrum of data availability*, in order to probe what data partitioning strategy will be preferred given different extents of data limitation. In addition, while heuristic splits are not plausible in our experiments, in future cases where applicable (e.g., text classification where the dataset can possibly be split heuristically based on the number of tokens or token types in the sentences), there is potential value in including such splits for more thorough comparisons.

Limitations

Our study faces two primary limitations. First, as described in Section 3.1, for each language, the data for experimentation come from the same domain, due to limited availability of datasets for morphological segmentation.

Second, since indigenous and endangered languages are often resource-constrained, after constructing each new test sample, we split the residual data into training and eval sets in order for the experimental setups to be consistent across languages, thereby not including a validation (or a tune) set for model parameter tuning. That said, the need for parameter tuning itself is an indication, to some extent, that the model may not generalize well to new data.

Ethics Statement and Broader Impact

Our study compares data partitioning strategies in order to better understand model generalizability, especially for indigenous languages. This research not only provides insights into conducting more reliable model evaluation in a broader sense, but also informs research for the development of more effective language technology for indigenous and endangered languages. Such advancements can contribute to language documentation efforts and support the respective speech communities. In addition, our selection of languages contributes to the ongoing efforts to promote language diversity in the field of natural language processing.

All original datasets used in our paper are publicly available, except for Hupa, which is an in-house dataset, derived with permission granted through academic relations as well as indirect relations with enthusiastic cooperation of the elders from the Hupa speech community. Therefore, ethical concerns of using the Hupa morphology data have been carefully considered.

Acknowledgements

We would like to thank Jordan Kodner for helpful feedback on earlier drafts of our paper. This material is based upon work supported, in part, by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001121C0186.

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

R Harald Baayen, Richard Piepenbrock, and Leon Gulikers. 1996. The CELEX lexical database (cd-rom).

Phyllis E. Wms. Bardeau. 2007. *The Seneca Verb: Labeling the Ancient Voice*. Seneca Nation Education Department, Cattaraugus Territory.

Emily Bender. 2019. The #BenderRule: On naming the languages we study and why it matters. *The Gradient*.

Emily M. Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Gabriela Caballero. 2008. *Choguita Rarámuri (Tarahumara) phonology and morphology*. Ph.D. thesis, University of California, Berkeley.

Gabriela Caballero. 2010. Scope, phonology and morphology in an agglutinating language: Choguita Rarámuri (Tarahumara) variable suffix ordering. *Morphology*, 20:165–204.

Ann Clifton and Anoop Sarkar. 2011. Combining morpheme-based machine translation with post-processing morpheme prediction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 32–42, Portland, Oregon, USA. Association for Computational Linguistics.

Michael Collins. 2002. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 1–8. Association for Computational Linguistics.

Ryan Cotterell, Thomas Müller, Alexander Fraser, and Hinrich Schütze. 2015. Labeled morphological segmentation with semi-Markov models. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 164–174, Beijing, China. Association for Computational Linguistics.

Jeremiah Curtin. 1888-1889. *Hupa vocabulary December 1888-January 1889*. National Anthropological Archives: NAA MS 2063.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Fanny Ducel, Karën Fort, Gaël Lejeune, and Yves Lepage. 2022. Do we name the languages we study? The #BenderRule in LREC and ACL articles. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June 2022*, pages 564–573. European Language Resources Association.

Elodie Gauthier, Laurent Besacier, Sylvie Voisin, Michael Melese, and Uriel Pascal Elingui. 2016. Collecting resources in sub-Saharan African languages for automatic speech recognition: a case study of Wolof. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3863–3867, Portorož, Slovenia. European Language Resources Association (ELRA).

Kyle Gorman and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2786–2791, Florence, Italy. Association for Computational Linguistics.

Katharina Kann, Jesus Manuel Mager Hois, Ivan Vladimir Meza-Ruiz, and Hinrich Schütze. 2018. Fortification of neural morphological segmentation models for polysynthetic minimal-resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 47–57, New Orleans, Louisiana. 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.

Alfred Kroeber. 1900-1906. *Untitled Hupa text*. Transcription in Kroeber's hand included in Goddard (1903-1906), notebook #4.

Mikko Kurimo, Mathias Creutz, Matti Varjokallio, Ebru Arsoy, and Murat Saraclar. 2006. Unsupervised segmentation of words into morphemes-morpho challenge 2005 application to automatic speech recognition. In *Ninth International Conference on Spoken Language Processing*, pages 1021–1024.

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

Viet-Bac Le and Laurent Besacier. 2009. Automatic speech recognition for under-resourced languages: application to Vietnamese language. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(8):1471–1482.

Zoey Liu, Robert Jimerson, and Emily Prud'hommeaux. 2021. Morphological segmentation for Seneca. In *Proceedings of the First Workshop on Natural Language Processing for Indigenous Languages of the Americas*, pages 90–101, Online. Association for Computational Linguistics.

Zoey Liu, Justin Spence, and Emily Prud'hommeaux. 2023. Investigating data partitioning strategies for crosslinguistic low-resource ASR evaluation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 123–131, Dubrovnik, Croatia. Association for Computational Linguistics.

Manuel Mager, Özlem Çetinoğlu, and Katharina Kann. 2020. Tackling the low-resource challenge for canonical segmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5237–5250, Online. Association for Computational Linguistics.

Manuel Mager, Arturo Oncevay, Elisabeth Mager, Katharina Kann, and Thang Vu. 2022. BPE vs. morphological segmentation: A case study on machine translation of four polysynthetic languages. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 961–971, Dublin, Ireland. Association for Computational Linguistics.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Justin Mott, Ann Bies, Stephanie Strassel, Jordan Kodner, Caitlin Richter, Hongzhi Xu, and Mitchell Marcus. 2020. Morphological segmentation for low resource languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3996–4002, Marseille, France. European Language Resources Association.

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.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Anders Søgaard, Sebastian Ebert, Jasmijn Bastings, and Katja Filippova. 2021. We need to talk about random splits. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1823–1832, Online. Association for Computational Linguistics.

Sebastian Spiegler, Andrew van der Spuy, and Peter A. Flach. 2010. Ukwabelana - an open-source morphological Zulu corpus. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1020–1028, Beijing, China. Coling 2010 Organizing Committee.

Alonso Vasquez, Renzo Ego Aguirre, Candy Angulo, John Miller, Claudia Villanueva, Željko Agić, Roberto Zariquiey, and Arturo Oncevay. 2018. Toward Universal Dependencies for Shipibo-Konibo. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 151–161, Brussels, Belgium. Association for Computational Linguistics.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Mary F. Woodward. 1953. *Survey of California and Other Indian Languages*. University of California Berkeley, Woodward.002.

A Notes on computing time and infrastructure

All experiments are run on a research computing cluster. The models applied here are open-source (Section 3.4). Given a new test sample size and a data partitioning strategy, with CRF, the total computing time for training the models from differently sized training sets as well as generating predictions for the evaluation sets ranges from less than a minute for Hupa, to around 20m for Zulu; with each of the seq2seq models, the total computing time spans from 6h30m for Hupa, to one day and a half for Zulu. All models are trained in sequential order with a single GPU with 8GB of memory.

B Detailed Results

This appendix contains Table 7-8.

C Regression results

Regression coefficients of other control variables are presented in Table 9: (1) the method for deriving new test samples; (2) morpheme overlap; (3) the relative ratios between the training and the eval sets for the number of words; (4) the average number of morphemes per word; (5) and the average number of morpheme types per word.

Language	Data partitioning strategy	CRF	LSTM	TRANSFORMER_TINY	TRANSFORMER
Mexicanero	random	0.85 (*0.02*)	0.69 (*0.05*)	0.75 (*0.04*)	0.58 (*0.04*)
	adversarial	0.85 (*0.02*)	0.67 (*0.06*)	0.74 (*0.04*)	0.57 (*0.04*)
Nahuatl	random	0.75 (*0.03*)	0.62 (*0.04*)	0.65 (*0.04*)	0.51 (*0.04*)
	adversarial	0.74 (*0.03*)	0.6 (*0.04*)	0.63 (*0.04*)	0.49 (*0.04*)
Yorem Nokki	random	0.80 (*0.02*)	0.70 (*0.04*)	0.73 (*0.03*)	0.61 (*0.04*)
	adversarial	0.79 (*0.02*)	0.68 (*0.04*)	0.72 (*0.03*)	0.59 (*0.04*)
Wixarika	random	0.76 (*0.02*)	0.7 (*0.03*)	0.69 (*0.03*)	0.59 (*0.03*)
	adversarial	0.76 (*0.02*)	0.68 (*0.03*)	0.68 (*0.03*)	0.58 (*0.03*)
Raramuri	random	0.75 (*0.03*)	0.63 (*0.05*)	0.69 (*0.03*)	0.52 (*0.08*)
	adversarial	0.74 (*0.03*)	0.61 (*0.05*)	0.67 (*0.03*)	0.50 (*0.07*)
Popoluca	random	0.88 (*0.02*)	0.55 (*0.05*)	0.60 (*0.04*)	0.50 (*0.04*)
	adversarial	0.88 (*0.02*)	0.54 (*0.05*)	0.58 (*0.03*)	0.49 (*0.03*)
Tepehua	random	0.79 (*0.03*)	0.50 (*0.04*)	0.49 (*0.04*)	0.40 (*0.03*)
	adversarial	0.78 (*0.03*)	0.49 (*0.04*)	0.46 (*0.09*)	0.39 (*0.04*)
Shipibo-Konibo	random	0.85 (*0.02*)	0.50 (*0.04*)	0.53 (*0.04*)	0.42 (*0.03*)
	adversarial	0.84 (*0.02*)	0.48 (*0.04*)	0.52 (*0.04*)	0.41 (*0.03*)
Seneca	random	0.95 (*0.01*)	0.92 (*0.03*)	0.91 (*0.03*)	0.78 (*0.01*)
	adversarial	0.95 (*0.01*)	0.92 (*0.01*)	0.90 (*0.02*)	0.76 (*0.01*)
Hupa	random	0.78 (*0.03*)	0.57 (*0.05*)	0.60 (*0.04*)	0.46 (*0.06*)
	adversarial	0.77 (*0.03*)	0.56 (*0.05*)	0.58 (*0.04*)	0.45 (*0.06*)
English	random	0.76 (*0.02*)	0.65 (*0.05*)	0.66 (*0.10*)	0.53 (*0.04*)
	adversarial	0.75 (*0.02*)	0.64 (*0.05*)	0.67 (*0.05*)	0.52 (*0.04*)
German	random	0.73 (*0.02*)	0.66 (*0.04*)	0.64 (*0.06*)	0.53 (*0.04*)
	adversarial	0.72 (*0.02*)	0.64 (*0.04*)	0.63 (*0.04*)	0.53 (*0.03*)
Finnish	random	0.74 (*0.03*)	0.60 (*0.04*)	0.59 (*0.04*)	0.46 (*0.03*)
	adversarial	0.73 (*0.03*)	0.59 (*0.05*)	0.58 (*0.05*)	0.45 (*0.03*)
Turkish	random	0.74 (*0.02*)	0.65 (*0.04*)	0.68 (*0.07*)	0.58 (*0.03*)
	adversarial	0.73 (*0.02*)	0.64 (*0.03*)	0.68 (*0.02*)	0.56 (*0.02*)
Indonesian	random	0.90 (*0.01*)	0.86 (*0.02*)	0.88 (*0.05*)	0.68 (*0.03*)
	adversarial	0.89 (*0.01*)	0.86 (*0.02*)	0.87 (*0.04*)	0.66 (*0.03*)
Zulu	random	0.85 (*0.01*)	0.84 (*0.01*)	0.79 (*0.01*)	0.64 (*0.05*)
	adversarial	0.85 (*0.01*)	0.83 (*0.01*)	0.78 (*0.01*)	0.63 (*0.05*)
Akan	random	0.82 (*0.01*)	0.76 (*0.02*)	0.67 (*0.15*)	0.70 (*0.02*)
	adversarial	0.81 (*0.01*)	0.75 (*0.02*)	0.69 (*0.14*)	0.70 (*0.02*)
Swahili	random	0.77 (*0.02*)	0.68 (*0.10*)	0.68 (*0.11*)	0.57 (*0.12*)
	adversarial	0.76 (*0.02*)	0.68 (*0.09*)	0.66 (*0.12*)	0.57 (*0.11*)
Telugu	random	0.71 (*0.02*)	0.64 (*0.04*)	0.68 (*0.03*)	0.53 (*0.04*)
	adversarial	0.69 (*0.02*)	0.59 (*0.11*)	0.64 (*0.09*)	0.46 (*0.10*)

Table 7: Language-by-language $F1$ scores averaged across differently sized new test samples, given each combination of a data partitioning strategy, test set proportion, and model architecture. Here new test samples are generated **randomly**. Standard deviations are (*italicized*).

Language	Data partitioning strategy	CRF	LSTM	TRANSFORMER_TINY	TRANSFORMER
Mexicanero	random	0.55 (*0.20*)	0.51 (*0.16*)	0.5 (*0.16*)	0.41 (*0.14*)
	adversarial	0.46 (*0.21*)	0.43 (*0.18*)	0.42 (*0.18*)	0.36 (*0.15*)
Nahuatl	random	0.52 (*0.14*)	0.44 (*0.12*)	0.44 (*0.09*)	0.35 (*0.07*)
	adversarial	0.45 (*0.15*)	0.37 (*0.11*)	0.38 (*0.08*)	0.32 (*0.07*)
Yorem Nokki	random	0.53 (*0.16*)	0.42 (*0.12*)	0.44 (*0.12*)	0.34 (*0.10*)
	adversarial	0.46 (*0.14*)	0.38 (*0.11*)	0.42 (*0.10*)	0.32 (*0.07*)
Wixarika	random	0.68 (*0.07*)	0.61 (*0.10*)	0.58 (*0.08*)	0.50 (*0.06*)
	adversarial	0.64 (*0.07*)	0.56 (*0.08*)	0.52 (*0.08*)	0.46 (*0.03*)
Raramuri	random	0.52 (*0.12*)	0.4 (*0.12*)	0.45 (*0.12*)	0.36 (*0.09*)
	adversarial	0.46 (*0.10*)	0.33 (*0.06*)	0.39 (*0.08*)	0.31 (*0.07*)
Popoluca	random	0.80 (*0.05*)	0.39 (*0.11*)	0.42 (*0.09*)	0.36 (*0.10*)
	adversarial	0.78 (*0.07*)	0.35 (*0.09*)	0.40 (*0.07*)	0.34 (*0.08*)
Tepehua	random	0.68 (*0.10*)	0.39 (*0.09*)	0.39 (*0.07*)	0.37 (*0.07*)
	adversarial	0.63 (*0.11*)	0.34 (*0.07*)	0.35 (*0.06*)	0.32 (*0.05*)
Shipibo-Konibo	random	0.64 (*0.10*)	0.32 (*0.08*)	0.33 (*0.08*)	0.27 (*0.06*)
	adversarial	0.53 (*0.17*)	0.25 (*0.08*)	0.26 (*0.09*)	0.21 (*0.07*)
Seneca	random	0.89 (*0.05*)	0.82 (*0.09*)	0.79 (*0.07*)	0.59 (*0.05*)
	adversarial	0.86 (*0.04*)	0.78 (*0.08*)	0.75 (*0.05*)	0.56 (*0.04*)
Hupa	random	0.67 (*0.11*)	0.50 (*0.14*)	0.48 (*0.11*)	0.39 (*0.11*)
	adversarial	0.63 (*0.10*)	0.45 (*0.11*)	0.45 (*0.08*)	0.36 (*0.10*)
English	random	0.52 (*0.05*)	0.41 (*0.05*)	0.43 (*0.04*)	0.31 (*0.04*)
	adversarial	0.48 (*0.08*)	0.35 (*0.08*)	0.37 (*0.07*)	0.27 (*0.05*)
German	random	0.63 (*0.10*)	0.54 (*0.10*)	0.52 (*0.09*)	0.43 (*0.08*)
	adversarial	0.59 (*0.09*)	0.49 (*0.07*)	0.47 (*0.08*)	0.38 (*0.07*)
Finnish	random	0.63 (*0.07*)	0.49 (*0.07*)	0.45 (*0.05*)	0.34 (*0.04*)
	adversarial	0.60 (*0.08*)	0.46 (*0.07*)	0.40 (*0.06*)	0.31 (*0.04*)
Turkish	random	0.73 (*0.07*)	0.57 (*0.09*)	0.58 (*0.06*)	0.46 (*0.06*)
	adversarial	0.69 (*0.09*)	0.52 (*0.09*)	0.53 (*0.06*)	0.43 (*0.07*)
Indonesian	random	0.60 (*0.19*)	0.53 (*0.21*)	0.56 (*0.19*)	0.39 (*0.20*)
	adversarial	0.42 (*0.24*)	0.36 (*0.24*)	0.37 (*0.25*)	0.23 (*0.12*)
Zulu	random	0.78 (*0.06*)	0.73 (*0.07*)	0.69 (*0.05*)	0.49 (*0.04*)
	adversarial	0.74 (*0.06*)	0.68 (*0.08*)	0.68 (*0.05*)	0.49 (*0.05*)
Akan	random	0.77 (*0.14*)	0.67 (*0.13*)	0.67 (*0.14*)	0.61 (*0.10*)
	adversarial	0.68 (*0.22*)	0.58 (*0.16*)	0.62 (*0.15*)	0.55 (*0.12*)
Swahili	random	0.71 (*0.11*)	0.66 (*0.11*)	0.67 (*0.11*)	0.58 (*0.11*)
	adversarial	0.65 (*0.09*)	0.63 (*0.10*)	0.62 (*0.11*)	0.55 (*0.10*)
Telugu	random	0.44 (*0.08*)	0.40 (*0.10*)	0.43 (*0.09*)	0.33 (*0.09*)
	adversarial	0.37 (*0.07*)	0.32 (*0.06*)	0.35 (*0.06*)	0.26 (*0.06*)

Table 8: Language-by-language $F1$ scores averaged across differently sized new test samples, given each combination of a data partitioning strategy, test set proportion, and model architecture. Here new test samples are generated **adversarially**. Standard deviations are (*italicized*).

Language	**Randomly generating new test samples**	**Morph overlap**	**N of words**	**Avg. N of morph per word**	**Avg. N of morph type per word**	R^2
Mexicanero	0.51***	0.56***	1.42***	3.05***	-4.19***	0.87
Nahuatl	0.69***	0.98***	0.58***	2.45***	-2.09***	0.90
Yorem Nokki	0.52***	0.87***	-0.54***	0.93***	0.54**	0.82
Wixarika	-0.16***	-0.40***	-0.72***	-2.74***	2.99***	0.72
Raramuri	0.59***	1.22***	0.79***	2.57***	-2.78***	0.87
Popoluca	0.57***	0.68***	-0.29***	0.03	0.33*	0.94
Tepehua	0.41***	0.66***	-0.56***	-0.72***	1.30***	0.92
Shipibo-Konibo	0.98***	1.29***	0.83***	2.91***	-2.64***	0.95
Seneca	3.66***	1.29***	3.33***	4.27***	-4.48***	
Hupa	0.47***	0.95***	0.33***	1.53***	-1.72***	0.92
English	0.86***	1.17***	0.66***	2.00***	-1.72***	0.92
German	-0.02	0.72***	-0.84***	-0.54***	1.57***	0.89
Finnish	0.65***	0.79***	0.94***	2.00***	-2.03***	0.93
Turkish	0.85***	0.71***	1.25***	2.62***	-3.02***	0.86
Indonesian	1.18***	1.14***	0.09	2.20***	-0.51*	0.93
Zulu	1.96***	1.16***	2.10***	3.98***	-4.22***	0.90
Akan	1.92***	1.43***	1.90***	3.61***	-4.11***	0.59
Swahili	0.60***	0.89***	0.59*	1.89***	-1.78**	0.51
Telugu	0.58***	0.64***	0.14	1.43***	-0.77**	0.84

Table 9: Regression coefficients of other control variables and R^2 of the regression model for each language; the number of * suggests significance level: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$.