Analogy Models for Neural Word Inflection

Ling Liu and **Mans Hulden**
University of Colorado
first.last@colorado.edu

Abstract

Analogy is assumed to be the cognitive mechanism speakers resort to in order to inflect an unknown form of a lexeme based on knowledge of other words in a language. In this process, an analogy is formed between word forms within an inflectional paradigm but also across paradigms. As neural network models for inflection are typically trained only on lemma-target form pairs, we propose three new ways to provide neural models with additional source forms to strengthen analogy-formation, and compare our methods to other approaches in the literature. We show that the proposed methods of providing a Transformer sequence-to-sequence model with additional analogy sources in the input are consistently effective, and improve upon recent state-of-the-art results on 46 languages, particularly in low-resource settings. We also propose a method to combine the analogy-motivated approach with data hallucination or augmentation. We find that the two approaches are complementary to each other and combining the two approaches is especially helpful when the training data is extremely limited.

1 Introduction

Morphological tasks such as the task of morphological inflection generation have attracted great research interest in recent years. SIGMORPHON has organized annual shared tasks on morphological inflection in the past five years (Cotterell et al., 2016; Cotterell et al., 2017a; Cotterell et al., 2018; McCarthy et al., 2019; Vylomova et al., 2020). In the typical SIGMORPHON shared task of morphological inflection, a lemma (citation form) and a morphosyntactic description (MSD) consisting of a set of features are provided, and the task is to generate an inflected form for the lemma corresponding to the MSD.

Neural network models have been very successful in handling natural language processing (NLP) problems, and have achieved new state of the arts in almost every area of NLP, including character-level sequence to sequence transformation tasks like morphological inflection, especially when there are abundant labeled data (Goldberg, 2016). However, neural network models are usually very data-hungry, and the performance of such models can suffer when labeled data is limited. Unfortunately, the large amounts of labeled data needed is not always available and can be difficult to obtain for many languages.

As interest has grown in low-resource NLP, several effective strategies to improve the performance of neural models have surfaced, and neural network models have become the dominant approach in low-resource settings as well. Such efforts for the morphological inflection task include engineering the neural network architecture to take better advantage of linguistic knowledge (Aharoni and Goldberg, 2017; Wu et al., 2018; Wu and Cotterell, 2019; Canby et al., 2020), designing data hallucination techniques to generate synthetic data based on existing labeled data (Silfverberg et al., 2017; Bergmanis et al., 2017; Anastasopoulos and Neubig, 2019; Yu et al., 2020), augmenting the training data by making better use of labeled or unlabeled data (Kann and Schütze, 2017; Kann et al., 2017a; Silfverberg et al., 2018; Silfverberg and Hulden, 2018; Liu and Hulden, 2020), and cross-lingual transfer learning, i.e. using labeled data in related languages to train models for the target language (McCarthy et al., 2019).

Proceedings of the 28th International Conference on Computational Linguistics, pages 2861–2878
Barcelona, Spain (Online), December 8-13, 2020

ID	MSD	Lexeme1 *"draw"*	Lexeme2 *"sweep"*	Lexeme3 *"carry over the head"*	Lexeme4 *"wear"*	Lexeme5 *"ask"*
1	V;NFIN	guhit	walis	sunong	suot	tanong
2	V;AGFOC;LGSPEC1	magguguhit	magwawalis	magsusunong	magsusuot	magtatanong
3	V;IPFV;AGFOC	nagguguhit	nagwawalis	nagsusunong	nagsusuot	nagtatanong
4	V;PFV;AGFOC	nagguhit	nagwalis	nagsunong	nagsuot	nagtanong
5	V;PFOC;LGSPEC1	guguhitin	wawalisin	susunungin	susuutin	tatanungin
6	V;IPFV;PFOC	ginuguhit	winawalis	sinusunong	sinusuot	tinatanong
7	V;PFV;PFOC	ginuhit	winalis	sinunong	sinuot	tinanong

Table 1: Tagalog paradigm examples

Model architecture engineering and data hallucination and augmentation techniques have seen consistent performance gains in current literature, but the effect of cross-lingual transfer for morphological inflection is less consistent. Some work has shown advances by conducting cross-lingual learning (Kann et al., 2017b; Anastasopoulos and Neubig, 2019; Murikinati and Anastasopoulos, 2020; Scherbakov, 2020; Peters and Martins, 2020), while some others have not found obvious improvements (Bergmanis et al., 2017; Rama and Çöltekin, 2018; Çöltekin, 2019; Hauer et al., 2019; Madsack and Weißgraeber, 2019).

Wu et al. (2020) shows the success of the Transformer architecture (Vaswani et al., 2017) for character-level transduction tasks, as is also supported by the results of the SIGMORPHON 2020 shared task 0 on morphological inflection (Vylomova et al., 2020). One approach the SIGMORPHON 2020 shared task 0 participating teams adopted to tackle the low-resource languages is data augmentation. The winning system (Liu and Hulden, 2020) reorganized the shared task data into partial paradigms and augmented the training data by inflecting from multiple known source forms in a paradigm—as opposed to the prevailing practice of just using the lemma form. This turns out to be very effective, and the system achieves the best performance in average accuracy and Levenshtein edit distance. Other participating systems (Yu et al., 2020; Singer and Kann, 2020; Murikinati and Anastasopoulos, 2020; Scherbakov, 2020) and the baseline system show the positive effect of data hallucination, which has also been evidenced by previous studies (Silfverberg et al., 2017; Anastasopoulos and Neubig, 2019). This motivates us to explore the following three questions:

1. Can one improve upon the choice of source forms to use in generating an inflected form?
2. Is data hallucination complementary to augmenting training data by using multiple source forms, or are the two strategies orthogonal, particularly in low-resource scenarios?
3. Is ensembling or model selection of multiple models necessary for best results?

For the first question, we follow the practice of organizing individual inflectional examples into incomplete paradigms and propose different ways motivated by the analogy mechanism to make use of known forms in the same paradigm as well as *across* paradigms, which can achieve even better results. For the second question, we conduct an experiment to combine the previous strategy with a data hallucination approach, and find that the two approaches are complementary in general, and that using both approaches is especially helpful when the training data is extremely limited ($<$ 1,000 training examples). Comparison of results by different models as to training data size, paradigm completion rate, and language groups, did not find dominant advantage of any single model, indicating that model ensembling or model selection is worthwhile.[1]

2 Model descriptions

2.1 Motivation

Analogy is assumed to be at the core of human cognition and it is assumed to be the mechanism by which we can inflect an unknown word given the other word forms we know (Blevins and Blevins, 2009). For

[1] Our code are publicly available at `https://github.com/LINGuistLIU/Analogy_for_inflection`.

example, Table 1 presents paradigm examples from Tagalog. If we know that the imperfective aspect with agent focus (V;IPFV;AGFOC) form for the Tagalog verb guhit (*"draw"*) is nagguguhit (*"is drawing"*), we can predict the V;IPFV;AGFOC form of another Tagalog verb, walis (*"sweep"*), to be nagwawalis (*"is sweeping"*). In this process, the analogy happens between the inflected form and the lemma, i.e. between guhit and nagguguhit, and between walis and nagwawalis, where commonality is found in the stem part between pairs. This part of the analogy has attracted much explicit attention and discussion in literature. It is the mechanism the typical morphological inflection task relies on. Though the lemma form is usually prioritized in morphological tasks, it is not always the most useful source form to inflect other forms in the same paradigm. The notion of *principal parts* states that there is a subset of forms in each paradigm which provides enough information to inflect other slots in the same paradigm correctly. This subset of forms is called a paradigm's principal parts (Finkel and Stump, 2007). For example, for Tagalog verbs, different agent focus (i.e. AGFOC) forms are very informative for each other's inflection, the perfective (i.e. PFV) and imperfective (i.e. IPFV) forms of patient focus (i.e. PFOC) are good sources for each other to inflect from, but AGFOC and PFOC forms are not very reliable predictors for each other. In the Tagalog example, forms which are more closely related are reliable sources for each other. But this is not always the case in every language. In some other languages, the reliable source form for a target form may not be closely related. The linguistic notion of *Priscianic formation* generalizes the situation where a slot in an inflectional paradigm is reliably formed from another slot of the same paradigm which is not necessarily closely related (Haspelmath and Sims, 2013). Both the principal parts and the Priscianic formation notions go against the idea of prioritizing the lemma as the only source form and encourage the use of other slots in the paradigm as source forms to predict the target slot from. Previous work (Cotterell et al., 2017b; Kann et al., 2017a; Liu and Hulden, 2020) has attempted to incorporate the notion of principal parts into neural network models for morphological inflection.

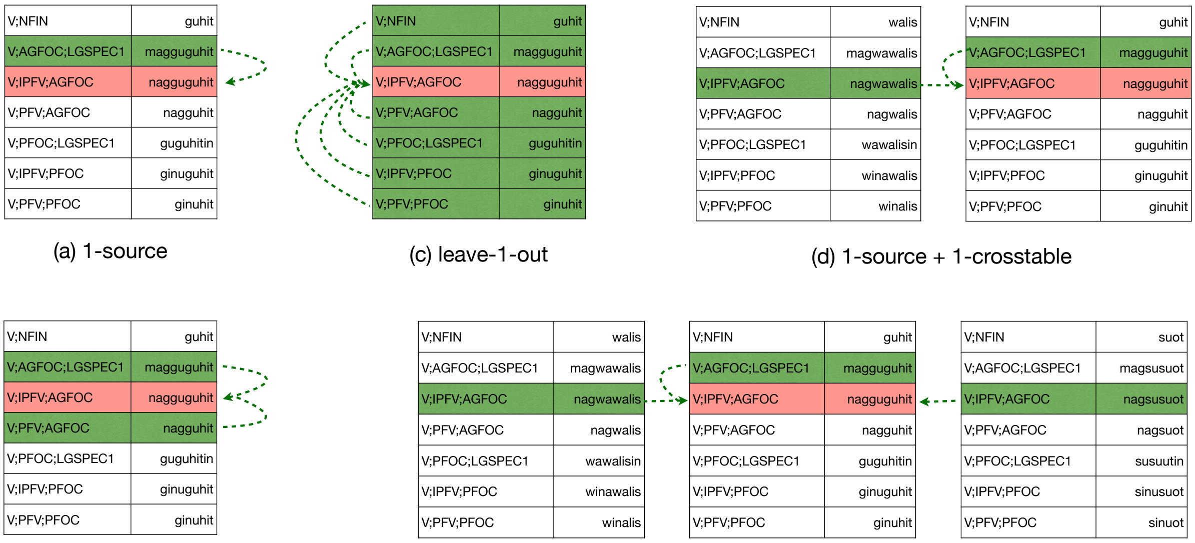
However, when we inflect walis V;NFIN → nagwawalis V;IPFV;AGFOC by analogy to guhit V;NFIN → nagguguhit V;IPFV;AGFOC, analogy also happens between paradigms: we also compare between guhit and walis, and between nagguguhit and nagwawalis, where commonality is found in the affix part between pairs. We resort to both the intraparadigmatic analogy and the interparadigmatic analogy in order to inflect unknown words from our knowledge of other words. There exists previous work trying to catch both parts of analogical reasoning (Hulden, 2014; Ahlberg et al., 2014; Ahlberg et al., 2015; Forsberg and Hulden, 2016; Silfverberg et al., 2018), though neural network models for morphological inflection have been relying on the neural model itself to catch the interparadigmatic analogy implicitly and haven't explicitly incorporated the cross-paradigm information.

Neural models for morphological inflection are traditionally trained to inflect from the lemma form only. In our Tagalog example, then, every form of the verb walis would be predicted from the NFIN form walis. Since models trained in this fashion perform quite well, they must have implicitly learned to form the analogies described above, even though only one source form is used. The root of our investigation, therefore, is the question: is it advantageous to explicitly provide source forms other than the lemma form when the model is trained?

2.2 Model architectures

Liu and Hulden (2020) convert the morphological inflection task into a partial paradigm completion problem, and use each form or each pair of forms together with the corresponding MSDs as input to the morphological inflection model of the Transformer architecture, which generates the inflected form for the target MSD (Figure 1 (a) and (b)). As this approach turned out to be very effective in modeling low-resource languages, it motivates us to explore additional ways to make use of the given data inspired by the analogy mechanism.

***1-source* and *2-source* models** Since Liu and Hulden (2020) presented their results as an ensemble, and did not analyze the model performance of using one source slot and using two source slots individually, we first reproduce their work and conduct an analysis of the two models they proposed: the *1-source* model (see Figure 1(a)) where each given slot in the paradigm is used as the input to predict the



(a) 1-source

(c) leave-1-out

(d) 1-source + 1-crosstable

(b) 2-source

(e) 1-source + 2-crosstable

Figure 1: Illustration of model architectures. Forms in green show the possible configuration of source forms used to inflect a target form in red. For model (a), every given slot is used for the target slot prediction respectively and thus we would get 6 *1-source* training input-output pairs out of the example partial paradigm, though only one such input-output pair is illustrated. For model (b), every pair of given slots are used for the target slot prediction and thus we would get 15 *2-source* training input-output pairs out of the example partial paradigm, though only one such pair is illustrated. For models (d) and (e), the crosstable forms are the inflected forms of the current target MSD from randomly picked partial paradigms where this form has been given.

missing slot from (i.e. `source form + source MSD + target MSD → target form`), and the *2-source* model (see Figure 1(b)) where each pair of given slots is used as the input to the inflection model for predicting the target form (i.e. `source form1 + source MSD1 + source form2 + source MSD2 + target MSD → target form`). This increases the amount of training data compared to the typical morphological inflection task data format `lemma + target MSD → target form`, since every given slot or every given slot pairs are used.

***Leave-1-out* model** The *1-source* and *2-source* models make use of the principal parts idea in an indirect way, by using average score or majority vote to pick out a final prediction from the multiple predictions for the target form. Is it possible for the neural network model to learn to pick out the subset of slots which are the principal parts? In order to explore this question, we propose the *leave-1-out* model (see Figure 1(c)) where the concatenation of all the known forms followed by their corresponding MSDs is input to the morphological inflection model to predict the target form, and the morphological inflection model is expected to learn to pick out the subset of slots which are the principal parts.

***1-source+1-crosstable* and *1-source+2-crosstable* models** Considering the analogy between paradigms, we propose the *1-source+1-crosstable* model and the *1-source+2-crosstable* model. In the *1-source+1-crosstable* model (see Figure 1(d)), we propose to use each given slot with its corresponding MSD concatenated with the inflected form of another randomly picked lemma for the target MSD as input to predict the target form for the target lemma, `source form + source MSD + inflected form from another random table for the target MSD + target MSD + target MSD → target form`.

In the *1-source+2-crosstable* model (see Figure 1(e)), we propose to use each given slot with its corresponding MSD concatenated with the inflected form of another two randomly picked lemmas for the target MSD as input to predict the target form for the target lemma, i.e. `source form +`

`source MSD + inflected form from another random table for the target MSD + target MSD + inflected form from a second random table for the target MSD + target MSD + target MSD` $\rightarrow$ `target form`. The linguistic intuition for using the target slots of two other lemmas is that this can provide additional analogy sources which may be helpful for the neural model to learn from.

***1-source+hallucination* model** The approach of using each individual form or each pair of forms in a paradigm to predict a target form is essentially a method to augment the training data, but this data augmentation approach is different from the data augmentation method of "hallucination," where synthetic "plausible" data are generated based on known labeled data and added to the training data for the morphological inflection model. Both augmentation by reformatting training data and data hallucination have produced improvement in neural model performance for morphological inflection in low-resource settings, but to our knowledge no work has analyzed whether the two data augmentation approaches are complementary to each other. Therefore, we propose the *1-source+hallucination* approach. We will use the *1-source* input format proposed by Liu and Hulden (2020) to create more training data examples from the given data, generate synthetic data based on the newly formatted training examples with the data hallucination method proposed by Anastasopoulos and Neubig (2019), and combine the newly formatted training data with the hallucination data to train the morphological inflection model.

Transformer As the Transformer architecture has been shown to be very successful in handling character-level string transduction tasks such as morphological inflection (Wu et al., 2020; Vylomova et al., 2020), we adopt the Transformer architecture for all the inflection models in our experiments.

3 Experiments

We evaluate the performance of all the models on the low-resource languages in the SIGMORPHON 2020 shared task 0 on morphological inflection (Vylomova et al., 2020). For our experiments, we regard languages with less than 5,000 training examples as low-resource. There are 46 such languages from 17 language groups in the SIGMORPHON 2020 shared task 0 data. The training and development data sets of the shared task are provided as triples of lemma, target form and target MSD, e.g. `jump V;PST jumped`. The test data is missing the target form, which the morphology inflection model is expected to predict. This dataset contains labeled data for 1 to 3 different parts of speech (POSs) depending on the language (nouns, verbs, and adjectives). We follow the same method as Liu and Hulden (2020) to reconstruct paradigms from the shared task data. Detailed statistics about the data for each language, including training data size, POSs, paradigm size for each POS, the number of paradigms per POS, and average paradigm completion rate as well as language group information are provided in Tables 5 and 6 in the Appendix B. The final number of training examples after the *1-source* and *2-source* transformation of the original training data is also provided in these tables. In this dataset, the development set is usually $1/7$ of the original training set size and the test set is usually $2/7$ of the original training data size.

The SIGMORPHON 2020 shared task 0 provides 2 types of neural baselines: a Transformer architecture applied at the character level (Wu et al., 2020) and a BiLSTM-based sequence-to-sequence architecture with exact hard monotonic attention (Wu and Cotterell, 2019). Each type of architecture is trained in four different ways with identical hyperparameters: training one model for each language with and without data hallucination, or training one model per language group with and without data hallucination. This results in 8 baseline models: *trm-single*, *trm-hal-single*, *trm-shared*, *trm-hal-shared*, and *mono-single*, *mono-hal-single*, *mono-shared*, *mono-hal-shared*. All the baseline models are trained with only lemma as the source form. Since we adopt the Transformer architecture for morphological inflection, our work focuses on the comparison with the Transformer baselines.

We use the implementation of the Transformer architecture in the Fairseq toolkit[2] (Ott et al., 2019), and set the hyperparameters equal to the SIGMORPHON shared task Transformer baselines, except that we use beam search rather than greedy search for decoding. Details on the hyperparameters and training heuristics used in the current paper is provided in Appendix A. We train one model with the Fairseq

[2] `https://fairseq.readthedocs.io/en/latest/`

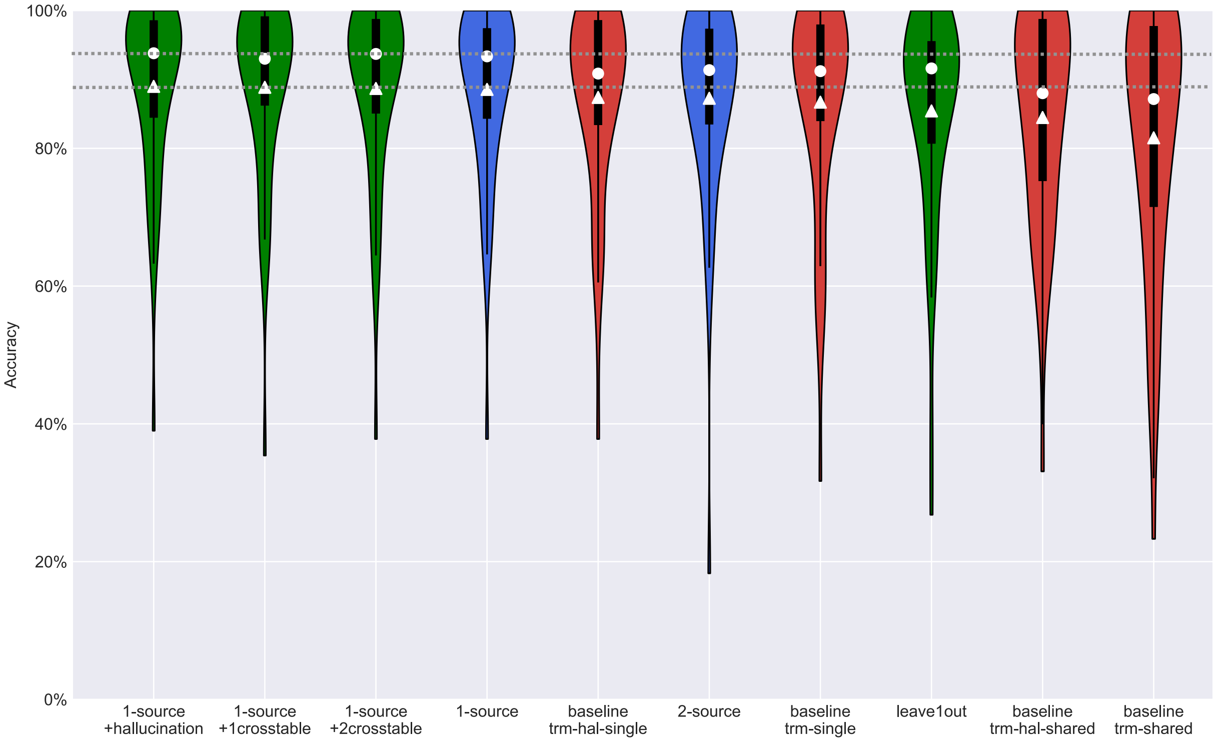


Figure 2: Comparison of model performance. The white circle indicates the median accuracy, and the white triangle indicates the average accuracy. The models are displayed from left to right with decreasing average accuracy. Figures in green are our proposed models, figures in blue are our reproduction results of the models proposed by Liu and Hulden (2020), and figures in red are produced from the official results for the Transformer baselines provided by the SIGMORPHON shared task organizers.

Transformer implementation with the same input-output format as the SIGMORPHON single-language baseline (i.e. *trm-single*) and the identical hyperparameters as we use for other model experiments. The result is presented in the row named *fairseq-trm-single* in Table 2. The *fairseq-trm-single* result is no better than the results for *trm-single* provided by the shared task organizers. This shows us that the improvements in performance in other models of our implementation truly reflects the contribution of incorporating more analogy sources to the input, and we can compare our results with the SIGMORPHON 2020 shared task 0 official baseline results.[3]

We reproduce the experiments with the *1-source* and *2-source* models on the 46 languages, and train models for our proposed models for comparison: *leave-1-out*, *1-source+1-crosstable*, *1-source+2-crosstable*, and *1-source+hallucination*. The evaluation metric we use to compare the performance of different models is accuracy, i.e. the fraction of correctly predicted target forms out of all predictions.

4 Results and discussion

Overall performance Figure 2 provides an overview of the performance by different models. Details about the accuracy of each language by each model is provided in Table 7 in Appendix B. We have the following findings based on the observation of overall model performance.

1. The good performance of the *1-source+1-crosstable* and the *1-source+2-crosstable* models supports the positive effect of providing more analogy sources for the neural model to learn from, or at least ones that differ from the citation form.

2. Data augmentation is necessary when the training data is limited. Models incorporating

[3] https://docs.google.com/spreadsheets/d/1ODFRnHuwN-mvGtzXA1sNdCi-jNqZjiE-i9jRxZCK0kg/edit#gid=258086389

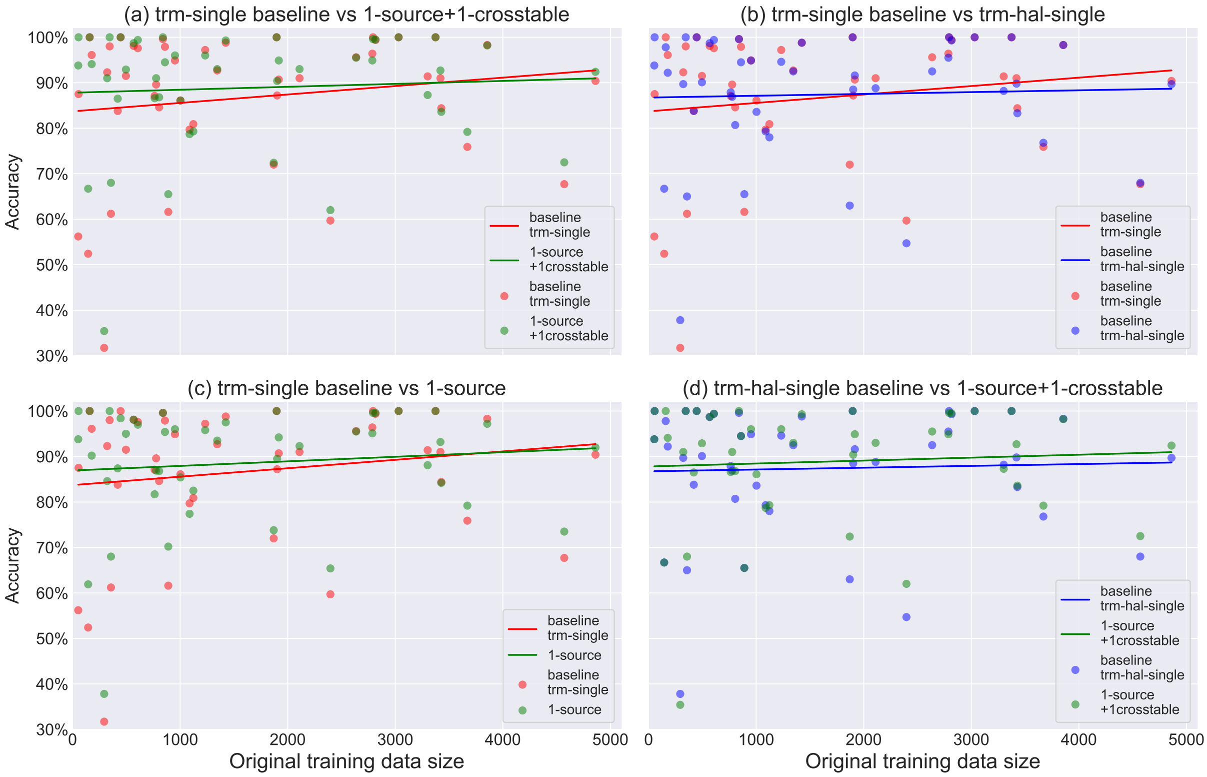


Figure 3: Scatterplots of original training size and accuracy for different models with regression lines.

data augmentation techniques in our experiments achieve better results. Specifically, the *1-source+hallucination* model produces the highest average accuracy, followed by the *1-source+1-crosstable* model and the *1-source+2-crosstable* model. The *1-source* model achieves an average accuracy higher than the baseline *trm-hal-single*. Though the *2-source* model has an average accuracy lower than baseline *trm-hal-single* and has a larger variance, its average accuracy is still higher than the baseline Transformer model trained without data hallucination, i.e. baseline *trm-single*.

3. The analogy-motivated approach of reformatting given data (Liu and Hulden, 2020) and the data hallucination approach (Anastasopoulos and Neubig, 2019) are complementary and can be profitably combined to improve the result. This is evidenced by the best performance of the *1-source+hallucination* model.

4. Our proposed *leave-1-out* model has the third lowest average accuracy, lower than the baseline *trm-single* model, indicating that the Transformer model failed to pick out the principal parts in our proposed way. The failure may be related to the limited amount of training data, which we leave to future work for validation.

5. The two baseline Transformer models trained per language group have the lowest average accuracies, indicating that the cross-lingual learning did not contribute to positive effects in these models.

Performance and data size Figure 3 plots the performance of the Transformer models with relation to the training data size. The regression lines indicate that reformatting the training data by adding more analogy sources for the model to learn from is essentially an effective data augmentation approach, but as data increases, the lines in plots (a) and (c) cross each other, indicating that this approach may not be necessary when there is abundant training data available. This is true for the data hallucination approach as well, as is shown in plot (b). Plot (d) shows that reformatting the data has similar effects as data hallucination, but that reformatting is in general more effective.

		0-1k	**1k-2k**	**2k-3k**	**3k-4k**	**4k-5k**	**0-5k**
	number of languages	21	10	6	7	2	46
Our results	leave-1-out	83.18	88.27	89.31	90.31	66.06	85.43
	1-source+1-crosstable	*87.95*	89.00	*90.75*	91.58	82.43	*88.85*
	1-source+2-crosstable	87.75	*89.36*	90.25	90.81	**83.17**	88.69
	1-source+hal	**88.24**	88.96	90.51	*91.75*	*82.84*	**88.99**
	1-source	87.09	88.96	**91.26**	91.71	82.76	88.55
	2-source	84.49	**89.51**	90.73	**91.93**	78.47	87.26
	fairseq-trm-single	83.40	87.61	90.03	90.92	78.68	86.12
Duplication of SIGMORPHON 2020 shared task0 baseline model results	trm-single	83.89	88.52	90.34	91.56	79.04	86.69
	trm-hal-single	86.90	86.99	88.47	90.91	78.85	87.39
	trm-shared	79.70	80.33	81.00	89.94	78.75	81.53
	trm-hal-shared	85.86	81.22	80.83	88.31	78.90	84.27
	mono-single	70.81	84.37	86.67	89.29	77.52	78.93
	mono-hal-single	83.82	84.83	87.28	89.59	78.10	85.12
	mono-shared	76.81	82.32	84.07	89.26	77.70	80.89
	mono-hal-shared	83.72	83.23	83.85	88.71	77.40	84.12

Table 2: Average accuracy (%) of each model grouped by training data size range. *fairseq-trm-single* is the transformer model we trained with the same hyperparameters as our other models with Fairseq implementation, for which the input is the same as *trm-single*. *1-source* and *2-source* rows present our reproduction results of the models proposed by Liu and Hulden (2020). SIGMORPHON 2020 shared task 0 baseline model results are copied from the published official results. The highest accuracies for each data size range are in boldface and the second highest is italicized.

We further break down the languages by training data size, and present the average accuracy for each data size range in Table 2 in order to explore whether any model show obvious advantage as to the amount of labeled data. Because the LSTM-based sequence-to-sequence architecture with exact hard monotonic attention model (Wu and Cotterell, 2019) was particularly designed to tackle low-resource languages, we include in the comparison the results for this type of models provided in the SIGMORPHON 2020 shared task 0 as well. Our findings are as follows:

1. For all the data size ranges, the models with additional analogy sources in the input (i.e. *1-source*, *2-source*, *1-source+1-crosstable*, *1-source+2-crosstable*, and *1-source+hallucination*) usually achieve better performance than models using only the lemma as input. This shows the effectiveness of explicitly providing more analogy sources as input to the neural morphological inflection model.

2. The *1-source+hallucination* model is effective across all data size ranges and especially for extremely low-resource scenarios. This model achieves significantly higher accuracy than other models for languages with fewer than 1,000 training examples, and its performance in other training data size ranges is also very close to the best models. This provides additional support to the benefit of combining the analogy-motivated data reformatting approach and the data hallucination approach.

3. The *2-source* model can be helpful in some scenarios, but it has high variance and is not as flexible and reliable as *1-source* models. The *2-source* model produces the highest average accuracy for languages with 1,000-2,000 or 3,000-4,000 training examples while its performance for languages with fewer than 1,000 or 4,000-5,000 training examples is much worse than the *1-source* models. This may be related to the fact that the *2-source* approach can augment data exponentially, which may result in a lot of pairs, misleading the model with noise. The *1-source* model has the highest

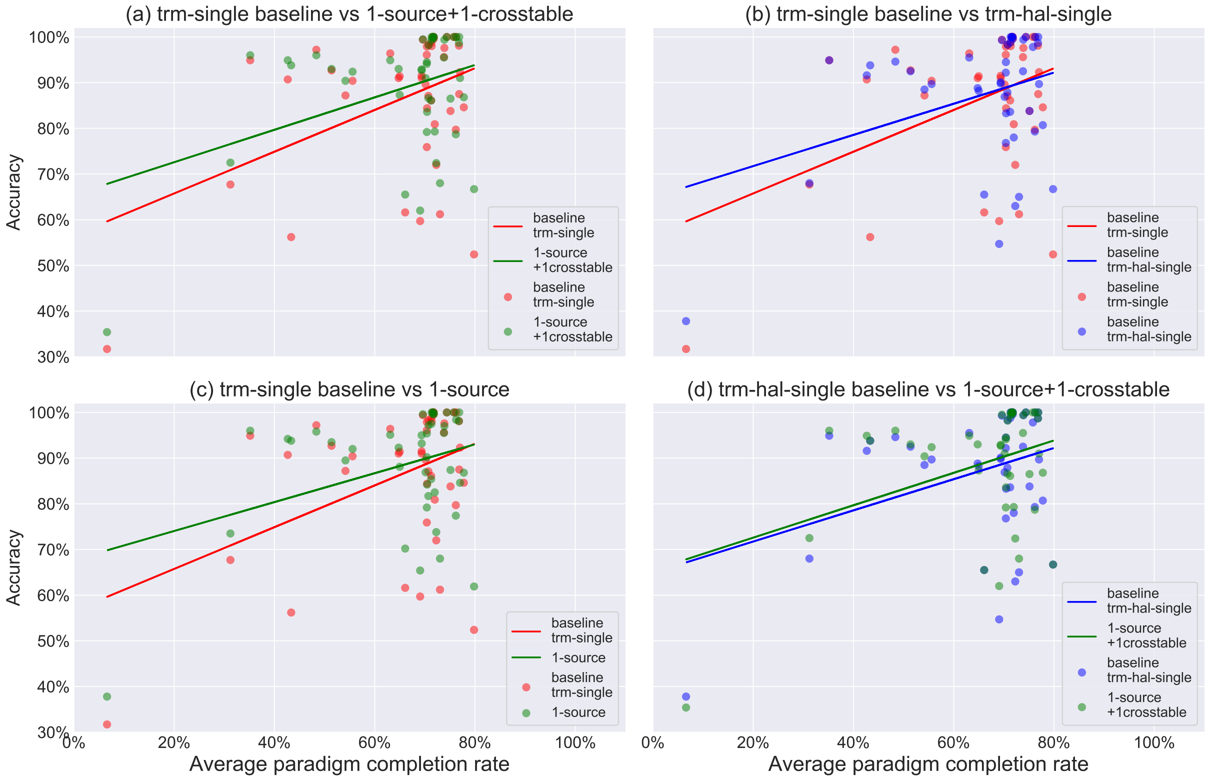


Figure 4: Scatterplots of average paradigm completion rate and accuracy for different models with regression lines.

average accuracy for languages with 2,000-3,000 training examples, and the *1-source+2-crosstable* model is the best one for the 4,000-5,000 range.

4. The *mono-hal-single* model usually produces higher accuracy than other models of the same architecture. However, it is still worse than most Transformer models. This reinforces earlier observation that the Transformer architecture is superior in handling character-level sequence transduction tasks over the hard-attention-enhanced LSTM encoder-decoder architecture (Wu et al., 2020).

5. Considering that no model shows obvious advantage across the board, model ensembling by picking the best model for each language on the development data set may be good practice in order to produce the best results for morphological inflection, as has been noted in Vylomova et al. (2020).

Performance and paradigm completion rate The relationship between the model performance and the paradigm completion rate is illustrated in Figure 4, from which we can see that languages with more known forms in paradigms tend to, on average, have higher accuracy. We also see that the contribution of data augmentation by either data reformatting with more analogy sources or data hallucination tends to decrease as the average paradigm completion rate increases. Still, data reformatting by analogy demonstrates the advantage over data hallucination, as reflected by the line for the *1-source+1-crosstable* model being above the regression line for the *trm-hal-single* model in plot (d) and the cross of the trend lines in plot (a) and plot (c) coming at a higher paradigm completion rate level than in plot (b).

Performance and language group The advantage of the strategy of reformatting data by the analogy mechanism is also observed across language groups, as is shown in Table 3, where the Siouan language group has an average accuracy in the shared task baseline results higher than the input augmented with analogy strategy methods, but this language group has only one language in our data, and the difference is not significant (higher by only 0.1%). However, none of the models show a general advantage across

lang	**lang**	**Our results**						**Copy of baseline results**			
group	**num**	**lv1out**	**1src+1**	**1src+2**	**1src+h**	**1src**	**2src**	**sing**	**h.sing**	**shrd**	**h.shrd**
AA	3	94.2	**96.7**	96.1	**96.7**	95.8	96.2	95.5	95.0	94.2	94.0
AL	1	40.1	72.5	**74.2**	73.6	73.5	73.0	67.7	68.0	67.7	68.0
AU	1	**92.3**	91.0	89.6	**92.3**	86.9	87.4	89.6	86.9	89.6	86.9
AO	5	83.5	**84.0**	83.0	83.5	83.4	82.7	81.2	81.6	82.8	79.9
DR	2	86.4	**87.6**	83.7	87.1	**87.6**	87.0	85.4	85.8	85.8	86.8
GE	4	83.6	85.4	**87.1**	86.6	**87.1**	85.4	83.2	84.2	67.2	74.1
IA	1	97.6	99.5	99.2	**99.6**	**99.6**	99.5	99.4	99.3	99.3	99.5
IR	2	92.9	**93.1**	93.0	92.9	92.9	88.8	73.3	91.8	79.3	91.8
NC	10	94.6	**98.2**	**98.2**	97.9	97.3	96.6	97.7	97.5	97.7	97.4
NS	1	93.8	**100.0**	**100.0**	**100.0**	**100.0**	87.5	87.5	**100.0**	87.5	**100.0**
OM	5	81.8	81.6	81.9	80.4	82.0	**83.1**	81.3	78.3	74.2	69.8
RO	1	62.8	94.1	94.1	**96.1**	90.2	82.4	**96.1**	92.2	76.5	84.3
ST	1	84.3	83.6	82.9	84.4	84.2	**84.5**	84.4	83.3	84.4	83.3
SI	1	95.1	95.5	95.3	95.1	95.5	93.9	**95.6**	92.5	**95.6**	92.5
TU	2	94.3	**99.1**	98.9	98.9	98.4	98.6	98.9	98.9	98.4	98.4
UR	5	69.5	74.2	73.7	**75.9**	73.6	70.0	72.5	74.0	45.7	65.9
UA	1	80.2	79.3	81.2	80.2	**82.5**	82.2	80.9	78.0	80.9	78.0

Table 3: Average accuracy (%) by language group. *AA*: Afro-Asiatic, *AL*: Algic, *AU*: Australian, *AO*: Austronesian, *DR*: Dravidian, *GE*: Germanic, *IA*: Indo-Aryan, *IR*: Iranian, *NC*: Niger-Congo, *NS*: Nilo-Saharan, *OM*: Oto-Manguean, *RO*: Romance, *ST*: Sino-Tibetan, *SI*: Siouan, *TU*: Turkic, *UR*: Uralic, *UA*: Uto-Aztecan. *lv1out*: leave-1-out, *1src+1*: 1-source+1-crosstable, *1src+2*: 1-source+2-crosstable, *1src+h*: 1-source+hallucination, *sing*, *h.sing*, *shrd*, *h.shrd* are results copied from SIGMORPHON 2020 shared task 0 results for the Transformer baseline models trained per language without (*sing*) or with (*h.sing*) data hallucination, or per language group without (*shrd*) or with (*h.shrd*) data hallucination.

language groups, again supporting the idea that model ensembling is a good choice for producing best results for a collection of diversified languages.

5 Conclusion

We propose three new ways to reformat training data using an analogy mechanism for morphological inflection in low-resource scenarios: *leave-1-out*, *1-source+1-crosstable*, *1-source+2-crosstable*. A systematic evaluation of the model performance shows that the proposed methods that provide both intra-paradigmatic and interparadigmatic analogy sources (i.e. *1-source+1-crosstable*, *1-source+2-crosstable*) are effective. In general, providing more analogy sources for the Transformer model to learn from is helpful. We further explore whether the data reformatting approach is orthogonal to data hallucination. Experimental results show that combining the two approaches is especially helpful when the training data is extremely limited. However, none of the models we evaluated in our experiments show an across-the-board advantage with respect to training data size, paradigm completion rate, or language groups, implying that model ensembling or model selection based on the development data is a good choice to achieve the best morphological inflection performance for a diversified collection of languages. This also indicates that morphological inflection generation is complicated, with many orthogonal factors affecting performance.

References

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2004–2015, Vancouver, Canada, July. Association for Computational Linguistics.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2014. Semi-supervised learning of morphological paradigms and lexicons. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 569–578, Gothenburg, Sweden, April. Association for Computational Linguistics.

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

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 984–996, Hong Kong, China, November. Association for Computational Linguistics.

Toms Bergmanis, Katharina Kann, Hinrich Schütze, and Sharon Goldwater. 2017. Training data augmentation for low-resource morphological inflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 31–39, Vancouver, August. Association for Computational Linguistics.

James P Blevins and Juliette Blevins. 2009. *Analogy in grammar: Form and acquisition*. Oxford University Press on Demand.

Marc Canby, Aidana Karipbayeva, Bryan Lunt, Sahand Mozaffari, Charlotte Yoder, and Julia Hockenmaier. 2020. University of Illinois submission to the 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 137–145, Online, July. Association for Computational Linguistics.

Çağrı Çöltekin. 2019. Cross-lingual morphological inflection with explicit alignment. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 71–79, Florence, Italy, August. 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, August. 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. 2017a. 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, August. Association for Computational Linguistics.

Ryan Cotterell, John Sylak-Glassman, and Christo Kirov. 2017b. Neural graphical models over strings for principal parts morphological paradigm completion. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 759–765, Valencia, Spain, April. 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, October. Association for Computational Linguistics.

Raphael Finkel and Gregory Stump. 2007. Principal parts and morphological typology. *Morphology*, 17(1):39–75.

Markus Forsberg and Mans Hulden. 2016. Learning transducer models for morphological analysis from example inflections. In *Proceedings of the SIGFSM Workshop on Statistical NLP and Weighted Automata (StatFSM)*, pages 42–50, Berlin, Germany, August. Association for Computational Linguistics.

Yoav Goldberg. 2016. A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420.

Martin Haspelmath and Andrea D Sims. 2013. *Understanding morphology*. Routledge.

Bradley Hauer, Amir Ahmad Habibi, Yixing Luan, Rashed Rubby Riyadh, and Grzegorz Kondrak. 2019. Cognate projection for low-resource inflection generation. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 6–11, Florence, Italy, August. Association for Computational Linguistics.

Mans Hulden. 2014. Generalizing inflection tables into paradigms with finite state operations. In *Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM*, pages 29–36. Association for Computational Linguistics.

Katharina Kann and Hinrich Schütze. 2017. Unlabeled data for morphological generation with character-based sequence-to-sequence models. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 76–81, Copenhagen, Denmark, September. Association for Computational Linguistics.

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

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017b. One-shot neural cross-lingual transfer for paradigm completion. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1993–2003, Vancouver, Canada, July. Association for Computational Linguistics.

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*, pages 153–161, Online, July. Association for Computational Linguistics.

Andreas Madsack and Robert Weißgraeber. 2019. AX semantics' submission to the SIGMORPHON 2019 shared task. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–5, Florence, Italy, August. Association for Computational Linguistics.

Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, Christo Kirov, Miikka Silfverberg, Sabrina J. Mielke, Jeffrey Heinz, Ryan Cotterell, and Mans Hulden. 2019. The SIGMORPHON 2019 shared task: Morphological analysis in context and cross-lingual transfer for inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–244, Florence, Italy, August. Association for Computational Linguistics.

Nikitha Murikinati and Antonios Anastasopoulos. 2020. The CMU-LTI submission to the SIGMORPHON 2020 shared task 0: Language-specific cross-lingual transfer. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 79–84, Online, July. 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, June. Association for Computational Linguistics.

Ben Peters and André F. T. Martins. 2020. One-size-fits-all multilingual models. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 63–69, Online, July. Association for Computational Linguistics.

Taraka Rama and Çağrı Çöltekin. 2018. Tübingen-Oslo system at SIGMORPHON shared task on morphological inflection. a multi-tasking multilingual sequence to sequence model. In *Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection*, pages 112–115, Brussels, October. Association for Computational Linguistics.

Andreas Scherbakov. 2020. The UniMelb submission to the 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 177–183, Online, July. Association for Computational Linguistics.

Miikka Silfverberg and Mans Hulden. 2018. 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, October-November. Association for Computational Linguistics.

Miikka Silfverberg, Adam Wiemerslage, Ling Liu, and Lingshuang Jack Mao. 2017. Data augmentation for morphological reinflection. In *Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection*, pages 90–99, Vancouver, August. Association for Computational Linguistics.

Miikka Silfverberg, Ling Liu, and Mans Hulden. 2018. A computational model for the linguistic notion of morphological paradigm. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1615–1626, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Assaf Singer and Katharina Kann. 2020. The NYU-CUBoulder systems for SIGMORPHON 2020 task 0 and task 2. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 90–98, Online, July. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

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, July. 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, July. Association for Computational Linguistics.

Shijie Wu, Pamela Shapiro, and Ryan Cotterell. 2018. Hard non-monotonic attention for character-level transduction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4425–4438, Brussels, Belgium, October-November. Association for Computational Linguistics.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2020. Applying the transformer to character-level transduction. *arXiv:2005.10213 [cs.CL]*, May.

Xiang Yu, Ngoc Thang Vu, and Jonas Kuhn. 2020. Ensemble self-training for low-resource languages: Grapheme-to-phoneme conversion and morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 70–78, Online, July. Association for Computational Linguistics.

A Hyperparameters

Here are the hyperparameter settings for our Transformer models. All the models share most of the hyperparameters as follows:

- UNK threshold = 1,
- encoder/decoder embedding dimension = 256,
- encoder/decoder hidden layer size = 1024,
- encoder/decoder number of layers = 4,
- encoder/decoder number of attention heads = 4,
- dropout = 0.3,
- batch size = 400,
- warmup update = 4000,
- learning rate = 0.001,
- label smoothing = 0.1,
- clip-norm = 1.0,
- optimization function: adam,
- adam-betas = (0.9, 0.98),
- activation function: ReLU,
- loss function: label smoothed cross entropy,
- beam search for generation with width of 5.

The frequency of checkpoint saving, the maximum number of parameter updates and early stop threshold vary between languages and models as stated below and summarized in Table 4, because the number of training data size varies after conversion and the setting is supposed to optimize the training process. The general pattern is for languages with more training data, the checkpoint is saved more frequently, the maximum number of updates is larger and more updates are allowed before early stop is enforced.

max updates	**save every 10 epochs**	**save every 1 epoch**	**update threshold for early stop**
20,000	**leave-1-out models**: all languages **1-source models**: (0, 20k) bod, ceb, ctp, czn, dak, dje, gaa, gmh, gml, gsw, hil, izh, kjh, kon, lin, lud, mao, mlg, mlt, ood, sot, syc, tel, tgk, tgl, vot, vro, xno, xty, zpv, zul		6,000
30,000	**2-source models**: (0, 40k) bod, ceb, czn, dak, dje, gaa, hil, kon, lin, mao, mlg, ood, sot, tgk, tgl, vro, xty, zpv, zul		
	1-source models: [20k, 200k) aka, ben, cly, frr, kan, kir, liv, lug, mwf, nya, orm, pus, sna, swa	**1-sources models**: [200k,) cre	15,000
50,000	**2-source model**: [40k, 200k) ctp, gmh, gml, gsw, izh, kjh, cly, mlt, orm, syc, tel, vot, xno	**2-source model**: [200k, 1m) lud, mwf, nya, sna	
100,000		**2-source model**: [1m, 10m) aka, ben, frr, kan,kir, liv, lug, pus, swa	20,000
500,000		**2-source model**: [10m,) cre	200,000

Table 4: Maximum number of updates, checkpoint saving frequency and early stop thresholds for different model and languages. Gray shaded cells are conditions which no languages follow.

B Data information and accuracy details

language	group	trn-raw	trn-1-src	trn-2-src	POS	psize	pnum	pfill-rate
tgk	Iranian	53	134	128	V	2	1	43.33
					N	4	74	
lud	Uralic	294	11,727	580,230	V	132	24	6.60
					N	36	100	
					ADJ	6	6	
gmh	Germanic	496	10,164	102,759	V	31	21	69.35
					N	9	8	
izh	Uralic	763	12,594	92,562	N	23	49	70.70
					ADJ	23	1	
gml	Germanic	890	8,672	53,802	V	20	39	66.07
					N	9	4	
					ADJ	23	9	
tel	Dravidian	952	9,722	52,299	V	25	116	35.16
					N	17	11	
ood	Uto-Aztecan	1,123	5,945	12,141	V	9	184	71.97
					N	3	189	
mlt	Afro-Asiatic	1,233	15,200	80,607	V	25	111	48.35
					N	9	1	
syc	Afro-Asiatic	1,917	22,899	145,317	N	27	155	42.65
					ADJ	13	29	
liv	Uralic	2,787	65,704	1,324,983	V	91	10	63.09
					N	19	143	
					ADJ	20	53	
ben	Indo-Aryan	2,816	76,100	1,038,630	V	42	84	69.61
					N	13	52	
kan	Dravidian	3,670	106,290	2,441,421	V	79	41	70.39
					N	11	124	
pus	Iranian	4,861	148,740	3,514,733	V	95	68	55.58
					N	9	270	
					ADJ	17	60	

Table 5: Data information for languages with more than one part-of-speech in the data. Languages are listed by the increasing order of the number of original training data. *trn-raw*: the amount of original training data, *trn-1-src*: the amount of training data after *1-source* conversion, which is the same for all *1-source* models, *trn-2-src*: the amount of training data after *2-source* conversion, *POS*: part-of-speech, *psize*: paradigm size, *pnum*: the number of paradigms, *pfill-rate*: paradigm completion rate in percentage.

language	group	trn-raw	trn-1-src	trn-2-src	POS	psize	pnum	pfill-rate
dje	Nilo-Saharan	56	182	131	V	4	27	76.85
mao	Austronesian	145	396	249	V	3	104	79.81
lin	Niger-Congo	159	648	734	V	5	57	75.79
xno	Romance	178	6,552	114,687	V	52	5	70.38
zul	Niger-Congo	322	1,550	2,407	V	6	87	77.01
sot	Niger-Congo	345	5,020	32,100	V	20	26	71.35
vro	Uralic	357	2,444	6,390	N	9	63	73.02
ceb	Austronesian	420	2,306	4,364	V	7	97	75.11
mlg	Austronesian	447	1,827	2,060	V	5	159	76.23
kon	Niger-Congo	568	2,351	2,726	V	5	200	76.8
gaa	Niger-Congo	607	4,662	13,785	V	10	95	73.89
mwf	Australian	777	20,078	253,983	V	38	29	70.15
zpv	Oto-Manguean	805	2,741	2,243	V	4	379	77.77
kjh	Turkic	840	10,448	55,644	N	17	75	71.76
hil	Austronesian	859	8,970	39,371	V	14	97	70.4
vot	Uralic	1,003	19,506	172,092	N	27	55	71.25
czn	Oto-Manguean	1,088	4,442	5,050	V	5	386	76.17
gsw	Germanic	1,345	14,688	70,272	V	20	145	51.38
orm	Afro-Asiatic	1,424	23,656	176,805	V	23	92	71.31
tgl	Austronesian	1,870	10,186	22,666	V	8	344	72.27
sna	Niger-Congo	1,897	44,544	488,046	V	31	86	74.38
frr	Germanic	1,902	64,188	1,112,916	V	66	51	54.16
xty	Oto-Manguean	2,110	10,161	15,393	V	7	594	64.77
ctp	Oto-Manguean	2,397	16,782	64,935	V	13	223	69.06
dak	Siouan	2,636	16,334	35,980	V	8	537	73.81
aka	Niger-Congo	2,793	85,702	1,253,043	V	42	96	71.65
nya	Niger-Congo	3,031	44,430	287,115	V	20	227	71.76
cly	Oto-Manguean	3,301	66,156	618,792	V	29	185	64.98
swa	Niger-Congo	3,374	122,218	2,117,934	V	50	97	71.57
lug	Niger-Congo	3,420	118,162	2,095,827	V	53	89	69.37
bod	Sino-Tibetan	3,428	13,065	13,683	V	5	1,335	70.44
kir	Turkic	3,855	156,650	3,051,111	V	57	98	70.75
cre	Algic	4,571	369,954	24,267,225	V	316	32	31.21

Table 6: Data information for languages with only one part-of-speech in the data. Languages are listed by the increasing order of the number of original training data. *trn-raw*: the amount of original training data, *trn-1-src*: the amount of training data after *1-source* conversion, which is the same for all *1-source* models, *trn-2-src*: the amount of training data after *2-source* conversion, *POS*: part-of-speech, *psize*: paradigm size, *pnum*: the number of paradigms, *pfill-rate*: paradigm completion rate in percentage.

		Our results						Copy of baseline results			
size	lang	lv1out	1src+1	1src+2	1src+h	1src	2src	sing	h.sing	shrd	h.shrd
$< 1k$	tgk	**93.8**	**93.8**	**93.8**	**93.8**	**93.8**	**93.8**	56.2	**93.8**	68.8	**93.8**
	dje	93.8	**100.0**	**100.0**	**100.0**	**100.0**	87.5	87.5	**100.0**	87.5	**100.0**
	mao	**66.7**	**66.7**	59.5	**66.7**	61.9	57.1	52.4	**66.7**	61.9	64.3
	lin	**100.0**	**100.0**	**100.0**	97.8	**100.0**	95.6	**100.0**	97.8	97.8	**100.0**
	xno	62.8	94.1	94.1	**96.1**	90.2	82.4	**96.1**	92.2	76.5	84.3
	lud	26.8	35.4	37.8	39.0	37.8	18.3	31.7	37.8	41.5	**62.2**
	zul	87.2	91.0	**92.3**	89.7	84.6	83.3	**92.3**	89.7	91.0	85.9
	sot	99.0	**100.0**	**100.0**	99.0	**100.0**	99.0	98.0	**100.0**	99.0	**100.0**
	vro	59.2	68.0	65.0	**71.8**	68.0	65.0	61.2	65.0	23.3	51.5
	ceb	84.7	86.5	86.5	82.0	**87.4**	86.5	83.8	83.8	86.5	82.9
	mlg	96.1	**100.0**	99.2	**100.0**	98.4	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	gmh	85.8	92.9	**95.0**	94.3	**95.0**	90.1	91.5	90.1	74.5	88.7
	kon	98.1	**98.7**	**98.7**	**98.7**	98.1	**98.7**	98.1	**98.7**	**98.7**	**98.7**
	gaa	73.4	99.4	99.4	**100.0**	97.0	98.2	97.6	99.4	**100.0**	**100.0**
	izh	82.1	86.6	84.8	87.0	81.7	86.2	87.0	**87.9**	56.3	70.1
	mwf	**92.3**	91.0	89.6	**92.3**	86.9	87.4	89.6	86.9	89.6	86.9
	zpv	**88.2**	86.8	86.8	84.6	86.8	85.5	84.6	80.7	81.6	84.2
	kjh	99.2	**100.0**	99.6	99.6	99.6	99.6	99.6	99.6	99.2	99.6
	hil	96.2	94.5	95.0	96.2	95.4	96.2	**97.9**	94.5	95.0	93.7
	gml	67.8	65.5	69.8	68.6	**70.2**	67.4	61.6	65.5	50.2	60.0
	tel	93.8	96.0	95.6	95.6	96.0	**96.3**	94.9	94.9	94.9	**96.3**
$< 2k$	vot	**86.1**	**86.1**	85.8	**86.1**	85.4	**86.1**	**86.1**	83.6	47.0	70.5
	czn	79.3	78.7	80.3	77.0	77.4	**80.7**	79.7	79.3	78.4	75.4
	ood	80.2	79.3	81.2	80.2	**82.5**	82.2	80.9	78.0	80.9	78.0
	mlt	91.5	96.0	94.6	95.5	95.8	94.6	**97.2**	94.6	93.5	93.5
	gsw	91.7	93.0	93.5	92.7	93.5	**94.0**	92.7	92.5	80.3	80.3
	orm	96.5	99.3	99.3	**99.8**	97.5	99.0	98.8	98.8	99.0	98.5
	tgl	73.6	72.4	**74.7**	72.8	73.8	73.6	72.0	63.0	70.5	58.4
	sna	**100.0**	**100.0**	99.6	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	frr	89.1	90.4	90.2	**90.6**	89.5	89.9	87.2	88.5	63.7	67.5
	syc	94.5	**94.9**	94.5	**94.9**	94.2	**94.9**	90.7	91.6	90.0	90.1
$< 3k$	xty	92.0	**93.0**	90.5	91.7	92.3	92.0	91.0	88.8	86.8	85.2
	ctp	62.2	62.0	62.7	61.2	65.4	**66.6**	59.7	54.7	44.1	32.9
	dak	95.1	95.5	95.3	95.1	95.5	93.9	**95.6**	92.5	**95.6**	92.5
	liv	93.3	94.9	94.9	95.5	95.1	94.3	**96.4**	95.5	60.3	75.2
	aka	95.7	99.6	98.8	**100.0**	99.6	98.2	**100.0**	**100.0**	99.9	99.7
	ben	97.6	99.5	99.2	**99.6**	**99.6**	99.5	99.4	99.3	99.3	99.5
$< 4k$	nya	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	cly	87.1	87.3	89.1	87.4	88.1	90.7	**91.4**	88.2	80.0	71.2
	swa	99.8	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	lug	92.5	92.7	93.6	**93.8**	93.2	92.9	91.0	89.8	90.9	89.4
	bod	84.3	83.6	82.9	84.4	84.2	**84.5**	84.4	83.3	84.4	83.3
	kan	79.1	**79.2**	71.8	78.6	**79.2**	77.7	75.9	76.8	76.7	77.2
	kir	89.4	98.2	**98.3**	98.2	97.2	97.7	**98.3**	**98.3**	97.6	97.1
$< 5k$	cre	40.1	72.5	**74.2**	73.6	73.5	73.0	67.7	68.0	67.7	68.0
	pus	92.0	**92.4**	92.2	92.1	92.0	84.0	90.4	89.7	89.8	89.8

Table 7: Accuracy (%) for each language by each of model we compare. The SIGMORPHON 2020 shared task 0 baseline results (i.e. the last 4 columns) are duplications of the shared task official results. The two columns shaded gray (i.e. *1src* and *2src*) are our results from Liu and Hulden (2020). Other models are our proposed methods. Languages are listed by the increasing order of the original training data size.