University of Illinois Submission to the SIGMORPHON 2020 Shared Task 0: Typologically Diverse Morphological Inflection

Marc E. Canby, Aidana Karipbayeva, Bryan J. Lunt,
Sahand Mozaffari, Charlotte R. Yoder, Julia Hockenmaier
University of Illinois at Urbana-Champaign
{marcec2,aidana2,bjlunt2,sahandm2,yoder6,juliahmr}@illinois.edu

Abstract

The objective of this shared task is to produce an inflected form of a word, given its lemma and a set of tags describing the attributes of the desired form. In this paper, we describe a transformer-based model that uses a bidirectional decoder to perform this task, and evaluate its performance on the 90 languages and 18 language families used in this task.

1 Introduction

The world's languages vary greatly in the richness and complexity of their morphological inflection systems. Indo-European languages such as Latin or German tend to inflect words by adding suffixes to a meaning-bearing root, while Austronesian languages like Malay or Tagalog use circumfixes to change the forms of nouns and verbs. It is important that Natural Language Processing (NLP) systems be able to generate inflected forms for a variety of languages, which can be used in downstream tasks such as language modeling or machine translation.

Task 0 of the SIGMORPHON 2020 Shared Task (Vylomova et al., 2020) encourages the development of morphological transduction models for a variety of the world's language families. Since the task features such a diverse set of languages, it is important to create a generalized model that is not overly biased toward certain language typologies.

In this paper, we present the University of Illinois submission to the task. We have modified the baseline transformer model (Wu et al., 2020) to use bidirectional decoding, following the work in Zhou et al. (2019). We believe the additional attention provided by the right-to-left decoding direction improves performance on many of the languages in the dataset. Our model outperforms the baseline transformer model on average rank and is among the best performing submissions for this year's task.

2 Task

The objective of Task 0 of the SIGMORPHON 2020 Shared Task (Vylomova et al., 2020) is to build a system that learns to generate morphological inflections. The model takes a lemma and a group of morphosyntactic tags as input and outputs the word inflected in the desired form. The following example comes from the German dataset:

predigen + V;IMP;SG;2
↓
predig

Here, we want to inflect *predigen* in the form specified by the tags V;IMP;SG;2, a 2nd person singular imperative verb. The desired output is *predig*.

2.1 Dataset

The organizers of the task provide datasets for 90 languages in total. 45 languages are treated as *development* languages – these languages span the Austronesian, Germanic, Niger-Congo, Oto-Manguean, and Uralic families, and were available for several months. The remaining 45 languages were released one week before the test sets and are considered *surprise* languages – they span 16 families, 13 of which are not represented by the development languages. The late release of these languages encourages the development of models that do not overly favor the development languages.

Each language has training and development files that consist of lemmata, morphosyntactic tags in the Unimorph Schema (Kirov et al., 2018), and inflected forms. A test set was released for each language one week before the deadline that contains only lemmata and morphosyntactic tags. The languages vary widely in the amount of data provided: for example, Finnish has approximately 100,000

Proceedings of the Seventeenth SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 137–145

https://doi.org/10.18653/v1/P17

training examples, while the Iranian language Tajik has only 53 training examples. This large disparity underscores the need for models that are not biased toward certain datasets or languages.

3 Method

3.1 Motivation

Recent work on morphological inflection has shown that an encoder-decoder framework using transformers produces state-of-the-art results (Wu et al., 2020). In our study, we have modified the baseline transformer model to use bidirectional *decoding* – that is, the prediction of a character is conditioned not only on the characters preceding it but also on those following it.

This approach is linguistically motivated, because it is common for an inflectional affix to be phonetically conditioned on the phonemes in its environment. For example, the underlying morpheme *ā* (a long *a*) marking the Latin present indicative can be expressed as the allomorph *a* (a short *a*) when followed by a stop consonant: *laudās* (2nd sg.) vs. *laudat* (3rd sg.). Kazakh exhibits regressive assimilation when adding the third person possessive suffix: the lemma *kitap* changes to *kitabı*. Here, the vowel in the suffix precipitates the voicing of the previous consonant.

It is standard to use bidirectional encoding to capture context in the source word (Wu and Cotterell, 2019; Wu et al., 2018), but we believe that a bidirectional decoder can better capture phonetic and orthographic dependencies in inflected forms. To our knowledge, no such method has been applied to a morphological transduction task before.

3.2 Previous Work

Neural models for morphological inflection have been studied extensively in previous SIGMORPHON Shared Tasks (Cotterell et al., 2017, 2018; McCarthy et al., 2019). Successful approaches include encoder-decoder frameworks using recurrent neural networks (RNN's) with attention (Cho et al., 2014; Wu and Cotterell, 2019; Wu et al., 2018). Hard monotonic attention has been particularly successful, due to the relatively rigid copy-like nature of inflection. Recent advances in the transformer architecture (Vaswani et al., 2017) have allowed transformer-based encoder-decoder models to become successful for inflection tasks as well (Wu et al., 2020). Indeed, the organizers provide us with two baselines: an RNN-based model with hard monotonic attention and a transformer baseline.

There has been some work on bidirectional decoding in the machine translation literature; however, we are unaware of any such work in morphological transduction tasks. Zhang et al. (2018) introduce an asynchronous bidirectional deocder based on RNN's; this approach first predicts the target sequence in reverse and then attends over this result to predict the target sequence left-to-right. Zhou et al. (2019) use a transformer model to predict both directions of the target sequence simultaneously, producing state-of-the-art results on translation tasks.

3.3 Model Architecture

Our model uses the technique of synchronous bidirectional decoding (Zhou et al., 2019). In this approach, the decoder pursues predictions of the inflected form in both the left-to-right (L2R) and right-to-left (R2L) directions simultaneously; that is, the first and last letters of the form are predicted first, then the second and second-to-last letters, and so on. At each step of decoding, each direction attends to the predictions of the other direction, so that an entire L2R prediction has been conditioned not only on itself but also on the R2L prediction. At inference time, the highest probability prediction in either direction is selected; it is reversed in the case that an R2L prediction has the highest probability.

In our implementation, the lemma and morphosyntactic tags are first embedded and encoded using the transformer-based encoder of the baseline. The decoder has been modified from the baseline in two ways. First, the decoder operates on previous L2R and R2L outputs in parallel at each time step. All weight matrices are shared between the two directions, and so this model has the same number of parameters as the baseline. Thus, the decoder makes both an L2R and an R2L prediction at each time step.

The second modification is the replacement of the multi-head intra-attention mechanism with a "Synchronous Bidirectional Attention" (SBAtt) mechanism, which allows each direction to attend to the opposite direction. The SBAtt mechanism is mostly the same as the standard intra-attention mechanism, except that the dot product attention has been replaced with "Synchronous Bidirectional Dot Product Attention". This can be summarized as follows:

$$\overrightarrow{H}^{history} = \text{Attention}\left(\overrightarrow{Q}, \overrightarrow{K}, \overrightarrow{V}\right)$$
$$\overrightarrow{H}^{future} = \text{Attention}\left(\overrightarrow{Q}, \overleftarrow{K}, \overleftarrow{V}\right)$$
$$\overrightarrow{H} = \text{Fusion}\left(\overrightarrow{H}^{history}, \overrightarrow{H}^{future}\right)$$

A similar equation holds for calculating $\overleftarrow{H}$. Here, Q, K, and V are the output hidden-state matrices of the previous layer, and the forward and backward arrows indicate the L2R and R2L matrices respectively. Zhou et al. (2019) provides three options for the Fusion function; given the empirical results of their study, we have used nonlinear interpolation in our implementation:

$$\overrightarrow{H} = (1-\lambda)\overrightarrow{H}^{history} + \lambda \tanh\left(\overrightarrow{H}^{future}\right)$$

We perform inference with a modified beam search. The algorithm tracks the k best L2R hypotheses and the k best R2L hypotheses. At each time step, the i^{th} best L2R hypothesis is paired with the i^{th} best R2L hypothesis, and these are fed to the decoder, which makes an L2R prediction and an R2L prediction. In the end, we select the hypothesis with the highest probability to length ratio; if an R2L hypothesis is selected, it is reversed before returning it.

3.4 Training & Model Configuration

Given training examples $\{x^{(i)}, y^{(i)}\}_{i=1}^{N}$, the model is trained to maximize the likelihood of the training data, accounting for both L2R and R2L probabilities:

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M}\Bigg[\log p\left(\overrightarrow{y}_j^{(i)} \middle| x^{(i)}, \overrightarrow{y}_{<j}^{(i)}, \overleftarrow{y}_{<j}^{(i)}; \theta\right) + \log p\left(\overleftarrow{y}_j^{(i)} \middle| x^{(i)}, \overleftarrow{y}_{<j}^{(i)}, \overrightarrow{y}_{<j}^{(i)}; \theta\right)\Bigg]$$

We train the model to minimize the negative log-likelihood loss function with label smoothing (Szegedy et al., 2016). We use an Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$. We employ a warmup-decay strategy for the learning rate as described in Vaswani et al. (2017) using 4000 warmup steps and initial learning rate of 0.001. Furthermore, special start-of-sentence tags $\langle l2r \rangle$ and $\langle r2l \rangle$ are used as the input to the decoder at the first step. A shared end-of-sentence token is used for both directions.

We keep most hyperparameters fixed for all languages in the dataset and train a separate model for each language. We use a batch size of 150, dropout of 0.3, embedding dimension of 256, maximum decoding length of 128, and gradient maximum ℓ_2 of 1.0. We tune the number of layers, the number of attention heads, the hidden dimension size, the label smoothing parameter λ_{smooth}, and the linear interpolation parameter for the Fusion function λ_{fusion}. The selection of these hyperparameters is described in Section 3.5.

Models were trained for 50,000 steps, or until accuracy on the development set flattened. In some cases, the accuracy curve was still rising, so some languages were trained to around 100,000 steps. We choose the model checkpoint with the highest development set accuracy to be used on the test data.

3.5 Hyperparameter Selection

We train a separate model for each language in the dataset and choose the hyperparameters by family. We perform a grid search for two languages in each family and select the best combination of hyperparameters based on accuracy on both of these languages. Where possible, we try to select two languages from different genera within a family, and in some families there is only one language present in the dataset. After selecting the optimal hyperparameters based on these results, we train individual models on each language in the family.

The hyperparameters we consider in our grid search are as follows:

- Num. Layers $\in \{4, 6\}$
- Num. Heads $\in \{4, 8\}$
- Hidden dimension $\in \{512, 1024\}$
- $\lambda_{smooth} \in \{0.0, 0.1\}$
- $\lambda_{fusion} \in \{0.1, 0.5\}$

We chose these hyperparameters because they appeared to cause variation in performance in our initial experiments. After tuning the development languages, it became clear that setting λ_{fusion} to 0.5 almost always degraded performance, and so this was left out of the hyperparameter search on the surprise languages. Setting $\lambda_{fusion} = 0.1$ is consistent with the experimental results in Zhou et al. (2019) on machine translation datasets. Table 3 in Appendix A.1 shows the hyperparameters chosen for each family.

There are some cases in which the languages used for hyperparameter tuning achieve better per-

Family	Accuracy			Edit-Distance		
	MONO	TRM	BI-TRM	MONO	TRM	BI-TRM
Afro-Asiatic	92.93	95.67	**96.37**	0.11	**0.05**	**0.05**
Algic	67.20	68.70	**70.30**	1.26	1.20	**1.16**
Australian	61.40	**90.00**	87.80	0.92	0.27	**0.26**
Austronesian	77.66	81.28	**82.30**	0.58	0.44	**0.41**
Dravidian	86.05	**87.10**	85.30	0.48	**0.46**	0.54
Germanic	86.88	**88.00**	87.38	0.30	**0.23**	0.25
Indo-Aryan	97.78	98.02	**98.18**	0.05	0.05	**0.04**
Iranian	63.00	82.50	**82.53**	1.04	**0.42**	0.46
Niger-Congo	97.72	97.72	**97.87**	0.04	0.04	**0.03**
Nilo-Sahan	0.00	87.50	**100.00**	2.88	0.19	**0.00**
Oto-Manguean	82.71	86.59	**87.49**	0.49	0.32	**0.28**
Romance	95.51	**99.25**	98.72	0.12	**0.02**	0.03
Sino-Tibetan	83.20	**84.40**	**84.40**	0.22	**0.20**	0.21
Siouan	92.90	**95.60**	94.90	0.16	**0.08**	0.10
Tungusic	55.30	**58.60**	58.30	1.20	**1.06**	1.09
Turkic	95.33	**95.96**	95.80	0.13	**0.10**	0.11
Uralic	83.21	**88.34**	88.18	0.39	0.29	**0.28**
Uto-Aztecan	76.30	80.80	**82.50**	0.49	0.41	**0.39**

Table 1: Macro-averages of accuracy and edit distance by language family. MONO refers to the hard monotonic baseline, TRM refers to the transformer baseline, and BI-TRM refers to our implementation using a bidirectional decoder.

formance with hyperparameters other than those selected for the family. In these cases, we used the best-performing hyperparameters found during the grid search. Table 4 in Appendix A.1 presents the hyperparameters used for these languages.

4 Experimental Results

Table 2 shows the number of languages on which our model is equal to or outperforms the baseline.

	Acc.		Avg. Edit Dist.	
	$\geq$	$>$	$\leq$	$<$
Development	27	18	30	14
Surprise	29	13	33	15

Table 2: The number of languages (out of 45) on which our model equals or outperforms ($\geq$ and $\leq$) or strictly outperforms ($>$ and $<$) the best of the two neural baseline models. It should be noted that on 5 of the development languages and 7 of the surprise languages, the baseline achieves perfect or near-perfect accuracy, making these languages impossible to outperform.

It is clear that by either metric, our model equals or outperforms the baseline on more than half of the languages, demonstrating that our model generally does not perform worse than the baseline.

Table 1 shows macro-averages of accuracy and edit distance by language family. For both metrics, our model outperforms the baseline transformer on 9 of the 18 language families and equals it on only one family. Interestingly, the two metrics do not agree on which families our model is best; when considering either metric, our model outperforms the baseline on 12 of the families.

Tables 5 and 6 in Appendix A.2 present full results on every language in the dataset. It is interesting to consider the L2R column, which indicates the percentage of test examples on which an L2R hypothesis was selected over an R2L hypothesis. There is considerable spread in the values of this column; this demonstrates that some languages strongly prefer one direction over the other, while others do not favor one direction in particular. It is important to remember that even though the inference algorithm returns only the best L2R or R2L hypothesis, the chosen direction is conditioned on the opposite direction; therefore, a language that appears to strongly prefer one direction may still gain important insight from the opposite direction.

5 Conclusion & Future Work

The promising results of our experiments demonstrate that some languages may be amenable to bidirectional decoding; however, more investigation is required to fully understand the merits of such an approach. For example, our results show that some languages strongly favor L2R or R2L hypotheses while others are less preferential. We would like to determine if there are particular linguistic features that make one direction more valuable than the other – for example, do inflected forms with suffixes prefer L2R decoding while in-

flected forms with prefixes favor R2L decoding? We propose performing this analysis by exploring correlations with linguistic features in the WALS database (Dryer and Haspelmath, 2013).

We would also like to investigate how often each direction produces the correct form, as well as the percentage of examples on which the two directions agree with each other. A high disagreement could indicate a higher value in one direction with respect to the other for a particular language. It would also be informative to compare the bidirectional decoding approach with a purely R2L transformer baseline, in addition to the L2R baseline provided by the organizers.

We also suspect that the bidirectional beam search algorithm can be improved if the hypotheses in one direction are paired with each of the hypotheses in the opposite direction when fed to the decoder at each time step. Furthermore, once the halfway-point of the target form is passed in the decoding, we should expect lots of overlap between the L2R and R2L forms. We would like to see if this information can be used to join the L2R and R2L predictions to produce a better inflected form.

In initial experiments we noticed that on some languages the bidirectional decoding model converges in considerably fewer epochs than the baseline transformer model, despite the same number of parameters. We want to fully investigate this phenomenon because, if it holds for many languages, it means that the model can gain insight more quickly with both directions than with just one.

Finally, in this work our models were trained from scratch on each individual language. We would like to investigate multilingual approaches by training separate models on individual language families or a single model for every involved language. In these ways, we hope to demonstrate the merits of bidirectional decoding and its implications for a morphological transduction task.

Acknowledgments

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) the State of Illinois, and as of December, 2019, the National Geospatial-Intelligence Agency. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications.

References

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

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

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

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian Mielke, Arya McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. UniMorph 2.0: Universal morphology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, Christo Kirov, Miikka Silfverberg, Sebastian 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. Association for Computational Linguistics.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J. Mielke, Shijie Wu, Edoardo Ponti, Rowan Hall Maudslay, Ran Zmigrod, Joseph Valvoda, Svetlana Toldova, Francis Tyers, Elena Klyachko, Ilya Yegorov, Natalia Krizhanovsky, Paula Czarnowska, Irene Nikkarinen, Andrej 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. 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*.

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

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2020. Applying the transformer to character-level transduction.

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. Association for Computational Linguistics.

Xiangwen Zhang, Jinsong Su, Yue Qin, Yang Liu, Rongrong Ji, and Hongji Wang. 2018. Asynchronous bidirectional decoding for neural machine translation. In *AAAI Conference on Artificial Intelligence*.

Long Zhou, Jiajun Zhang, and Chengqing Zong. 2019. Synchronous bidirectional neural machine translation. *Transactions of the Association for Computational Linguistics*, 7:91–105.

A Appendices

A.1 Hyperparameter Selection

In this section we present the hyperparameters used for each language. Tables 3 and 4 contain information about specific hyperparameter configurations for each family and for specific languages.

Family	Languages	# Layers	Hidden Size	# Heads	λ_{smooth}
Afro-Asiatic	Oromo Syriac	6	512	4	0.0
Algic	Cree	4	1024	4	0.0
Australian	Murrinh-Patha	6	512	4	0.1
Austronesian	Maori Tagalog	6	1024	4	0.1
Germanic	Old English Norwegian Bokmål	4	1024	8	0.0
Indo-Ayran	Sanskrit Bengali	4	512	8	0.0
Iranian	Persian Pashto	4	1024	4	0.0
Niger-Congo	Luganda Zulu	4	1024	4	0.0
Nilo-Sahan	Zarma	4	1024	4	0.0
Oto-Manguean	Yaitepec Chatino Chichimeca-Jonaz	4	1024	4	0.1
Romance	Asturian Ladin	6	512	8	0.0
Sino-Tibetan	Tibetan	6	1024	8	0.1
Siouan	Dakota	4	1024	8	0.0
Tungusic	Evenki	4	1024	4	0.1
Turkic	Kazakh Uyghur	4	1024	4	0.0
Uralic	Moksha Votic	4	512	4	0.0
Uto-Aztecan	O'odham	6	1024	4	0.1

Table 3: Selected hyperparameters by family. The "Languages" column indicates the languages we used for selecting the hyperparameters. The Dravidian family is not present, since it has exactly two languages; the hyperparameters for these languages can be seen in Table 4.

Family	Languages	# Layers	Hidden Size	# Heads	λ_{smooth}
Afro-Asiatic	Oromo	4	512	4	0.0
Austronesian	Tagalog	6	1024	8	0.0
Dravidian	Kannada	4	1024	8	0.1
Dravidian	Telugu	4	1024	4	0.0
Germanic	Old English	4	512	8	0.0
Oto-Manguean	Chichimeca-Jonaz	6	1024	8	0.1
Uralic	Votic	6	512	8	0.0

Table 4: Selected hyperparameters for certain languages on which we performed a grid search. These languages use different hyperparameters than their corresponding families, shown in Table 3, due to the fact that a more optimal configuration was discovered.

A.2 Complete Results Tables

In this section we show full results on each language.

Family	Language	Accuracy			Edit Distance			L2R
		MONO	TRM	BI-TRM	MONO	TRM	BI-TRM	
Austronesian	Cebuano	83.80	83.80	**87.40**	0.31	0.33	**0.26**	57.66
	Hiligaynon	92.40	**97.90**	96.60	0.22	**0.09**	0.10	26.47
	Maori	47.60	**52.40**	**52.40**	1.10	1.02	**0.95**	52.38
	Malagasy	99.20	**100**	**100**	0.01	**0**	**0**	9.45
	Tagalog	65.30	72.30	**75.10**	1.27	0.78	**0.73**	33.89
Germanic	Old English	75.80	**79.10**	78.40	0.44	**0.37**	0.38	77.41
	Danish	74.60	**76.30**	73.00	0.60	**0.25**	0.29	93.92
	German	**98.50**	97.70	98.00	0.06	0.03	**0.02**	95.18
	English	96.60	**96.90**	**96.90**	0.10	**0.06**	**0.06**	89.91
	North Frisian	86.10	**87.90**	87.60	0.40	**0.39**	0.42	54.30
	Middle High German	90.80	91.50	**92.90**	0.17	**0.11**	**0.11**	82.98
	Icelandic	97.10	97.00	**97.60**	0.06	0.07	**0.04**	88.79
	Dutch	98.90	99.00	**99.50**	0.02	0.02	**0.01**	79.48
	Norwegian Bokmål	76.90	**77.30**	74.80	0.47	**0.46**	0.51	95.20
	Swedish	98.80	98.70	**99.00**	0.08	**0.02**	**0.02**	92.04
Niger-Congo	Akan	**100**	**100**	99.90	**0**	**0**	**0.00**	67.23
	Gã	**100**	97.60	97.00	**0**	0.04	0.05	52.66
	Kongo	**98.70**	98.10	**98.70**	**0.01**	0.03	**0.01**	78.85
	Lingala	**100**	**100**	**100**	**0**	**0**	**0**	67.39
	Luganda	90.00	91.20	**92.80**	0.17	0.13	**0.11**	46.47
	Chewa	**100**	**100**	**100**	**0**	**0**	**0**	98.01
	Sotho	**100**	98.00	98.00	**0**	0.03	0.03	91.92
	Swahili	**100**	**100**	**100**	**0**	**0**	**0**	72.64
	Zulu	88.50	**92.30**	**92.30**	0.19	**0.13**	**0.13**	43.59
Oto-Manguean	San Pedro Amuzgos Amuzgo	93.50	94.70	**95.20**	0.17	0.13	**0.12**	21.12
	Eastern Highland Chatino	78.70	91.40	**91.80**	0.39	**0.15**	0.16	22.14
	Tlatepuzco Chinantec	89.00	91.60	**92.30**	0.16	**0.12**	**0.12**	64.64
	Yaitepec Chatino	45.90	61.20	**62.50**	2.28	1.00	**0.97**	63.71
	Zenzontepec Chatino	79.30	79.70	**84.60**	0.44	0.49	**0.33**	60.33
	Mezquital Otomi	**99.10**	99.00	**99.10**	**0.01**	**0.01**	**0.01**	32.13
	Sierra Otomi	97.90	**98.20**	98.00	0.06	**0.05**	**0.05**	82.91
	Chichimeca-Jonaz	**74.60**	74.50	74.20	0.59	0.60	**0.57**	63.96
	Yoloxóchitl Mixtec	90.70	91.00	**91.70**	0.23	0.22	**0.16**	69.33
	Chichicapan Zapotec	78.40	84.60	**85.50**	0.55	0.39	**0.32**	75.44
Uralic	Estonian	95.10	**95.60**	95.20	0.19	**0.17**	0.18	68.04
	Finnish	99.60	99.60	**99.70**	0.02	**0.01**	**0.01**	62.97
	Ingrian	68.80	87.10	**87.50**	0.60	0.24	**0.23**	86.16
	Karelian	99.30	99.30	**99.50**	0.04	**0.01**	**0.01**	50.79
	Livonian	92.50	**96.40**	95.50	0.13	**0.06**	0.07	52.24
	Moksha	92.80	**93.90**	93.60	0.24	**0.18**	0.19	81.11
	Meadow Mari	**93.30**	92.90	92.60	0.19	**0.15**	0.16	85.81
	Erzya	93.60	**94.50**	94.10	0.21	**0.17**	0.18	90.65
	Northern Sami	99.60	99.60	**99.70**	**0.01**	**0.01**	**0.01**	69.86
	Veps	82.70	**84.80**	83.30	0.45	**0.25**	0.27	84.56
	Votic	69.40	**86.10**	84.30	0.49	**0.21**	0.24	51.25

Table 5: Results for individual languages in the development language set. MONO refers to the hard monotonic baseline, TRM refers to the transformer baseline, and BI-TRM refers to our implementation using a bidirectional decoder. The L2R column shows the percentage of words in each language for which our model selects a left-to-right hypothesis as its final result. It should be noted that this column really indicates a "forwardness" percentage, as languages with a right-to-left orthography are processed in a right-to-left manner.

Family	Language	Accuracy			Edit Distance			L2R
		MONO	TRM	BI-TRM	MONO	TRM	BI-TRM	
Afro-Asiatic	Maltese	88.70	**97.20**	96.60	0.22	**0.05**	**0.05**	67.99
	Oromo	98.30	**99.00**	98.00	0.03	**0.02**	0.04	30.86
	Syriac	91.80	90.80	**94.50**	0.08	0.09	**0.06**	70.07
Algic	Cree	67.20	68.70	**70.30**	1.26	1.20	**1.16**	69.34
Australian	Murrinh-Patha	61.40	**90.00**	87.80	0.92	0.27	**0.26**	63.06
Dravidian	Kannada	77.30	**78.30**	76.10	0.70	**0.67**	0.76	60.15
	Telugu	94.80	**95.90**	94.50	0.25	**0.24**	0.32	42.49
Germanic	Middle Low German	60.60	**63.50**	58.40	1.03	**0.84**	1.10	52.16
	Swiss German	90.10	92.70	**93.20**	0.18	0.11	**0.10**	68.83
	Norwegian Nynorsk	84.60	86.40	**86.60**	0.24	0.21	**0.20**	85.76
Indo-Aryan	Bengali	98.80	99.40	**99.90**	0.03	0.05	**0.00**	93.54
	Hindi	**100**	**100**	**100**	**0**	**0**	**0**	75.07
	Sanskrit	92.90	**93.40**	**93.40**	0.16	0.15	**0.14**	65.77
	Urdu	**99.40**	99.30	**99.40**	**0.01**	**0.01**	**0.01**	88.62
Iranian	Persian	**100**	**100**	99.90	**0**	**0**	**0.00**	45.26
	Pashto	89.00	91.20	**91.40**	0.30	**0.25**	**0.25**	62.85
	Tajik	0.00	**56.30**	**56.30**	2.81	**1.00**	1.12	75.00
Niger-Congo	Shona	**100**	**100**	**100**	**0**	**0**	**0**	85.31
Nilo-Sahan	Zarma	0.00	87.50	**100**	2.88	0.19	**0**	43.75
Romance	Asturian	98.50	**99.40**	99.30	0.03	**0.01**	**0.01**	46.88
	Catalan	99.60	**99.80**	**99.80**	0.01	**0.00**	**0.00**	84.35
	Middle French	99.50	**99.80**	**99.80**	0.01	**0.00**	**0.00**	83.48
	Friulian	97.70	**99.80**	99.70	0.03	**0.00**	**0.00**	66.11
	Galician	99.70	**99.80**	**99.80**	**0.01**	**0.01**	**0.01**	80.65
	Ladin	99.00	**99.50**	**99.50**	0.02	**0.01**	**0.01**	61.38
	Venetian	99.50	**99.80**	99.70	0.01	0.01	**0.00**	52.60
	Anglo-Norman	70.60	**96.10**	92.20	0.82	**0.10**	0.18	60.78
Sino-Tibetan	Tibetan	83.20	**84.40**	**84.40**	0.22	**0.20**	0.21	37.39
Siouan	Dakota	92.90	**95.60**	94.90	0.16	**0.08**	0.10	71.20
Tungusic	Evenki	55.30	**58.60**	58.30	1.20	**1.06**	1.09	65.55
Turkic	Azerbaijani	79.50	**82.20**	81.90	0.42	**0.34**	**0.34**	87.70
	Bashkir	99.60	**99.80**	**99.80**	0.01	**0.00**	**0.00**	69.80
	Crimean Tatar	98.80	99.10	**99.30**	0.10	**0.01**	**0.01**	78.05
	Kazakh	97.40	97.90	**98.00**	0.15	0.12	**0.11**	63.46
	Kyrgyz	97.90	98.30	**98.80**	0.04	0.03	**0.02**	67.95
	Khakas	99.20	**99.60**	**99.60**	0.01	**0.00**	0.01	81.67
	Turkmen	86.50	**87.40**	85.60	0.45	**0.42**	0.50	82.09
	Uyghur	99.50	99.50	**99.70**	0.01	0.01	**0.00**	48.17
	Uzbek	99.60	**99.80**	99.50	**0.01**	**0.01**	0.02	67.58
Uralic	Komi-Zyrian	96.30	**96.90**	**96.90**	0.11	**0.07**	**0.07**	75.61
	Ludic	24.10	**32.90**	**32.90**	2.14	2.35	**2.13**	68.29
	Livvi	**94.50**	94.30	**94.50**	0.14	**0.09**	**0.09**	82.53
	Udmurt	97.80	**98.40**	**98.40**	0.06	**0.03**	**0.03**	74.02
	Võro	32.00	61.20	**63.10**	1.27	0.66	**0.62**	63.11
Uto-Aztecan	O'odham	76.30	80.80	**82.50**	0.49	0.41	**0.39**	62.42

Table 6: Results for individual languages in the surprise language set. MONO refers to the hard monotonic baseline, TRM refers to the transformer baseline, and BI-TRM refers to our implementation using a bidirectional decoder. The L2R column shows the percentage of words in each language for which our model selects a left-to-right hypothesis as its final result. It should be noted that this column really indicates a "forwardness" percentage, as languages with a right-to-left orthography are processed in a right-to-left manner.