The CMU-LTI submission to the SIGMORPHON 2020 Shared Task 0: Language-Specific Cross-Lingual Transfer

Nikitha Murikinati and Antonios Anastasopoulos
Language Technologies Institute
Carnegie Mellon University
nmurikin@andrew.cmu.edu, aanastas@cs.cmu.edu

Abstract

This paper describes the CMU-LTI submission to the SIGMORPHON 2020 Shared Task 0 on typologically diverse morphological inflection. The (unrestricted) submission uses the cross-lingual approach of our last year's winning submission (Anastasopoulos and Neubig, 2019), but adapted to use specific transfer languages for each test language. Our system, with fixed non-tuned hyperparameters, achieved a macro-averaged accuracy of 80.65 ranking 20th among 31 systems, but it was still tied for best system in 25 of the 90 total languages.

1 Introduction

Morphological inflection is the process that creates grammatical forms (typically guided by sentence structure) of a lexeme/lemma. As a computational task it is framed as mapping from the lemma and a set of morphological tags to the desired form, which simplifies the task by removing the necessity to infer the form from context. For an example from Asturian, given the lemma aguar and tags V;PRS;2;PL;IND, the task is to create the indicative voice, present tense, 2nd person plural form aguà.

Let $\mathbf{X} = x_1 \ldots x_N$ be a character sequence of the lemma, $\mathbf{T} = t_1 \ldots t_M$ a set of morphological tags, and $\mathbf{Y} = y_1 \ldots y_K$ be an inflection target character sequence. The goal is to model $P(\mathbf{Y} \mid \mathbf{X}, \mathbf{T})$. The problem has been studied in various settings through the SIGMORPHON shared tasks (Cotterell et al., 2016, 2017, 2018; McCarthy et al., 2019), with the 2019 edition focusing in particularly challenging low-resource scenarios. The 2020 edition (Vylomova et al., 2020) focused on generalization of systems across typologically diverse languages, regardless of data size.

In our submission we built upon our previous work (Anastasopoulos and Neubig, 2019), utilizing cross-lingual transfer from related languages, data hallucination, and a series of training techniques and regularizers. The defining change was that we attempted to create language-specific regimes for each test language, depending on the particular characteristics of the language, on the data availability for the particular test language and the availability of other related language data. As a result, for some high-resource languages we submitted systems without cross-lingual transfer, for some we used a single related high resource language, and for some we used multiple related languages. Last, for a few test languages we augmented our datasets with romanized versions of the training data, an approach that has shown promising results in concurrent work (Murikinati et al., 2020).

Our submissions are very competitive in 25 of the 90 test languages, with performance statistically significant similar to the best performing system, but fall behind in many other languages. We suspect that this is due to our not tuning of the system's hyperparameters towards higher-resource settings.

Proceedings of the Seventeenth SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 79–84
Online, July 10, 2020. ©2020 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17

Language	Accuracy	Language	Accuracy	Language	Accuracy	Language	Accuracy
aka	**99.1**	fas	96.2	lld	97.7	sna	**100.0**
ang	75.4	fin	97.3	lud	**53.7**	sot	**100.0**
ast	91.4	frm	98.8	lug	90.6	swa	**100.0**
aze	78.5	frr	85.5	mao	**69.0**	swe	95.4
azg	89.0	fur	98.3	mdf	92.7	syc	91.6
bak	97.4	gaa	**100.0**	mhr	90.8	tel	**94.9**
ben	98.6	glg	97.4	mlg	**100.0**	tgk	**93.8**
bod	**84.7**	gmh	90.1	mlt	88.7	tgl	64.0
cat	97.5	gml	**60.8**	mwf	70.3	tuk	85.4
ceb	**84.7**	gsw	84.9	myv	93.0	udm	97.5
cly	81.0	hil	92.4	nld	97.5	uig	91.9
cpa	83.5	hin	98.4	nno	74.2	urd	36.3
cre	44.9	isl	95.3	nob	**75.1**	uzb	51.5
crh	97.2	izh	80.8	nya	**100.0**	vec	98.8
ctp	50.2	kan	75.1	olo	91.5	vep	79.3
czn	**81.3**	kaz	88.5	ood	**79.0**	vot	77.2
dak	89.7	kir	88.4	orm	93.6	vro	**57.3**
dan	72.3	kjh	**98.8**	ote	97.0	xno	**90.2**
deu	92.8	kon	**98.1**	otm	97.4	xty	90.2
dje	**100.0**	kpv	95.9	pei	71.2	zpv	**82.9**
eng	96.5	krl	95.0	pus	68.6	zul	**89.7**
est	93.5	lin	**100.0**	san	92.6		
evn	55.0	liv	93.1	sme	97.9		

Table 1: Accuracy of our system on every language. We **highlight** the languages where our system was statistically equal to the best system (with $p < 0.005$).

2 System Description

Our system is the same as the one of Anastasopoulos and Neubig (2019): a neural multi-source encoder-decoder (which reads in the lemma and the tag sequences in a disentangled manner using two separate encoders) with a task-specific attention mechanism. We skip providing further redundant information and we direct the interested reader to (Anastasopoulos and Neubig, 2019) for all details. It is important to note, however, that we did not tune any model hyperparameters for our submissions (which we suspect contributed to the poor performance of our system in some languages); we used the default parameters from the system's distribution [1] which are tuned towards extremely low-resource settings.

Here, we provide an exhaustive list of modifications to the general pipeline that we devised for specific languages and language families.

Data Hallucination for tonal languages The data hallucination process of Anastasopoulos and Neubig (2019), inspired by Silfverberg et al. (2017), samples random characters from the language's alphabet to replace characters in *stem-like* regions discovered from the training examples through a simple alignment-based heuristic.

Tonal languages like Eastern Highland Chatino (cly), importantly, often denote the syllable's tone through superscript diacritics: take the Eastern Highland Chatino lemma `sqwe`^{14} and its second person singular number habitual mood inflected form `nsqwe`^{20}. The data hallucination technique would identify the substring `sqwe` as a stem-like region, and replace its characters with random ones. A completely random substitution, however, could lead to the creation of nonsensical syllables, if tone diacritics are inserted instead of letter characters e.g. if we hallucinated a `s`^{3}`ae`^{14} lemma for the above example. Similarly, if a stem-like region includes a tone diacritic, we would not want to randomly replace it with non-diacritic characters,

[1] https://github.com/antonisa/inflection

lest we end up with badly formed syllables without tone information.

To avoid these issues, we restrict the random substitutions for Oto-Manguean languages with tone diacritics, so that we only sample tone diacritics if we are substituting a tone diacritic (and similarly for letter characters). We have found this approach to significantly improve results in previous work on morphological inflection for Eastern Highland Chatino (Cruz et al., 2020).

Single-Language Systems for High Resource Languages For languages with more than 20,000 training examples, we decided to not use cross-lingual transfer nor data hallucination, as systems in previous SIGMORPHON shared tasks achieved very competitive performance on such high-resource settings without these additions. For languages with less than 20,000 but more than 10,000 training examples, we used our data hallucination process to create 10,000 additional training examples to be used for training.

Cross-Lingual Transfer from a Single Language For some languages we decided to use a single, high-resource related language to combine into our training to perform cross-lingual transfer, along with data hallucination. We based most these decisions in previous results (mainly from (Anastasopoulos and Neubig, 2019)), but some where our semi-arbitrary experimenter's intuitions. We provide a complete list of these settings:

- for Middle High German (gmh) we used German (deu),
- for Middle Low German (gml) we used German (deu) also bypassing data hallucination,
- for Swiss German (gsw) we used German (deu),
- for North Frisian (frr) we used Dutch (nld),
- for Kannada (kan) we used Telugu (tel),
- for Telugu (tel) we used Kannada (kan),
- for Asturian (ast) we used Galician (glg),
- for Friulian (fur) we used French (fra),
- for Ladin (lad) we used Friulian (fur),
- for Venetian (vec) we used Italian (vec),
- for Anglo-Norman (xno) we used Middle French (frm),
- for Azerbaijani (aze) we used Turkish (tur),
- for Khakas (kjh) we used Turkish (tur), but not including data hallucination, and
- for Võro (vro) we used Estonian (est).

Family	Sub-family	Acc.
Afro-Asiatic		91.3
	Semitic	90.1
Algic		44.9
Turkic		83.3
Austronesian		82.0
	Gr. Ctr. Philippines	80.4
Dravidian		85.0
IndoEuropean		87.5
	Germanic	84.3
	Romance	96.3
	Iranian	86.2
	Indic	81.5
Niger-Congo		97.7
	Bantoid	97.3
	Kwa	99.5
Oto-Manguean		82.4
	Zapotecan	73.9
	Otomian	97.2
Sino-Tibetan		84.7
Siouan		89.7
Songhay		100.0
Southern Daly		70.3
Uralic		86.7
	Mordvin	92.8
	Finnic	81.9
	Permic	96.7
Uto-Aztecan		79.0
Tungusic		55.0

Table 2: Results per language Family/Genus.

Multiple-Language Cross-Lingual Transfer We submitted systems with unique transfer language combinations for extremely low-resource languages for which several very related languages were available (all systems also included hallucinated data in the test language). Specifically:

- for Ingrian (izh) we used Estonian (est), Votic (vot), and a random sample (20,000 instances) from Finnish (fin) data,
- for Votic (vot) we used Estonian (est), Ingrian (izh), and a random sample (20,000 instances) from Finnish (fin) data,
- for Urdu (urd) we used Hindi (hin) and Bengali (ben),

- for Bashkir (bad) we used Turkish (tur), Kazakh (kaz), and Kyrgyz (kir),
- for Crimean Tatar (crh) we used Turkish (tur), Kazakh (kaz), and Kyrgyz (kir),
- for Kazakh (kaz) we used Turkish (tur), Bashkir (bad), and Kyrgyz (kir),
- for Kyrgyz (kir) we used Turkish (tur), Bashkir (bad), and Kazakh (kaz),
- for Uighur (uig) we used Turkish (tur) and Uzbek (uzb), and
- for Ludian (lud) we used 20,000 random samples from Karelian (krl) and Veps (vep).

Romanization for Different Scripts Last, we experimented with cross-lingual transfer *and* transliteration of related languages written in different script. The motivation lies in the observation made by Anastasopoulos and Neubig (2019) that often cross-lingual transfer results in smaller improvements if the transfer and the test language do not share the same script, even if the languages are related. They bring Arabic–Maltese and Kurmanji–Sorani as possible examples. In concurrent work (Murikinati et al., 2020) we experimented with transliterating the transfer language into the test language's script, with encouraging results in low-resource settings. Alternatively, if the training languages use the latin script but the test language does not, we found that that by romanizing the test language training data and concatenating them as another language (along with the data in the original script) also helped. We applied these strategies on the following language pairs.

Transliterating a transfer language into the test language's script:

1. for Maltese (mlt) we used Italian (ita) and romanized Hebrew (heb),
2. for Oromo (orm) we used romanized Arabic (ara) and romanized Hebrew (heb), and
3. for Bengali (ben) we used Sanskrit (san), Hindi (hin), and Sanskrit transliterated into the Bengali script using the Indic NLP library[2] (Kunchukuttan, 2020).

Romanizing the test language training data and training with both romanized and original, along with more romanized, related languages:

1. for Classical Syriac (syc) we used romanized Arabic (ara) and romanized Hebrew (heb), as well as romanized Classical Syriac (Classical Syriac originally uses a distinct script),
2. for Pashto (pus) we used romanized Farsi (fas) and romanized Pashto, while
3. for Tajik (tgk) we used romanized Farsi (fas) and romanized Tajik.

3 Results

Table 1 lists the accuracy of our submitted system in every language. We also report results per language family and genus in Table 2, to further facilitate an equitable evaluation across language families. Our system achieves a macro-averaged accuracy of 86.6% with a standard deviation of 14.3. Even though it does not use self-attention and we did not tune any hyper-parameters, our system still achieved competitive performance, tying for first in 25 of the 90 total languages (it still however does not outperform the best baseline system (Wu et al., 2020)).

These include languages that were generally easy for all systems, such as the Austronesian and the Niger-Congo ones. However, they also include the extremely low-resource languages like Ludian (lud), Võro (vro), and Middle Low German (gml), where we suspect that our system performed en par with the more sophisticated (and we suspect, tuned) systems due to our informed selection of languages for cross-lingual transfer.

The two languages where our system performs the worst are Algic (Cree) and Tungusic (Evenki). We suspect this is due to the fact that the data hallucination technique, which is crucial for such low resource settings, is not appropriate for capturing the vowel harmony of Evenki along with its agglutinating morphological patterns – the hallucinated data do not follow these patterns and hence do not guide the model towards learning them. As for Cree, we suspect that the problem lies again in the data hallucination process: the polysynthetic *and* fusional nature of Cree verb inflected forms is too complicated to be modeled by the simple character-level alignment model which is the first step for hallucination.

4 Conclusion and Future Work

The performance of our system in the 2020 SIGMORPHON Shared Task leaves many questions unanswered and several avenues to explore in future work. Regarding the choice of languages to use for cross-lingual transfer, we will further in-

[2] https://github.com/anoopkunchukuttan/indic_nlp_library

vestigate the use of automatic suggestion systems such as the one of Lin et al. (2019). With regards to modeling, we will update our model to use sparsemax (Martins and Astudillo, 2016), which can facilitate exact search and hopefully lead to better results (Peters and Martins, 2019).

As we anticipate and hope the shared task and the whole community will become more multilingual in the future, in the future we will employ the language/task selection method of Xia et al. (2020), which will allow us to tune the systems in a small subset of languages that will generalize well in all others. Similarly, we will employ more sophisticated techniques for learning in multilingual settings, such as differential data selection (Wang et al., 2019, 2020) which will allow us to optimize a single model to multiple model objectives (namely, each target language).

Acknowledgments

This material is based upon work generously supported by the National Science Foundation under grant 1761548.

References

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. In *Proc. EMNLP*, Hong Kong.

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 *Proc. CoNLL–SIGMORPHON*.

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 *Proc. CoNLL SIGMORPHON*.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared task—morphological reinflection. In *Proc. SIGMORPHON*.

Hilaria Cruz, Antonios Anastasopoulos, and Gregory Stump. 2020. A resource for studying chatino verbal morphology. In *Proc. LREC*. To appear.

Anoop Kunchukuttan. 2020. The indicnlp library.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135, Florence, Italy. Association for Computational Linguistics.

Andre Martins and Ramon Astudillo. 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proc. ICML*, pages 1614–1623.

Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, 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.

Nikitha Murikinati, Antonios Anastasopoulos, and Graham Neubig. 2020. Transliteration for cross-lingual morphological inflection. In *Proc. SIGMORPHON*. To appear.

Ben Peters and André FT Martins. 2019. It–ist at the sigmorphon 2019 shared task: Sparse two-headed models for inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 50–56.

Miikka Silfverberg, Adam Wiemerslage, Ling Liu, and Lingshuang Jack Mao. 2017. Data augmentation for morphological reinflection. *Proc. SIGMORPHON*.

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*.

Xinyi Wang, Hieu Pham, Paul Michel, Antonios Anastasopoulos, Graham Neubig, and Jaime Carbonell. 2019. Optimizing data usage via differentiable rewards. arXiv:1911.10088.

Xinyi Wang, Yulia Tsvetkov, and Graham Neubig. 2020. Balancing training for multilingual neural machine translation. In *Annual Conference of the Association for Computational Linguistics (ACL)*.

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

Mengzhou Xia, Antonios Anastasopoulos, Ruochen Xu, Yiming Yang, and Graham Neubig. 2020. Predicting performance for natural language processing tasks. In *Proc. ACL*. To appear.