

Neural Morphology Dataset and Models for Multiple Languages, from the Large to the Endangered

Mika Hämäläinen, Niko Partanen, Jack Rueter and Khalid Alnajjar

Faculty of Arts

University of Helsinki

firstname.lastname@helsinki.fi

Abstract

We train neural models for morphological analysis, generation and lemmatization for morphologically rich languages. We present a method for automatically extracting substantially large amount of training data from FSTs for 22 languages, out of which 17 are endangered. The neural models follow the same tagset as the FSTs in order to make it possible to use them as fallback systems together with the FSTs. The source code¹, models² and datasets³ have been released on Zenodo.

1 Introduction

Morphology is a powerful tool for languages to form new words out of existing ones through inflection, derivation and compounding. It is also a compact way of packing a whole lot of information into a single word such as in the case of the Finnish word *hatussanikinko* (in my hat as well?). This complexity, however, poses challenges for NLP systems, and in the work concerning endangered languages, morphology is one of the first NLP problems people address.

The GiellaLT infrastructure (Moshagen et al., 2014) has HFST-based (Lindén et al., 2013) finite-state transducers (FSTs) for several morphologically rich (and mostly Uralic) languages. These FSTs are capable of lemmatization, morphological analysis and morphological generation of different words.

These transducers are at the core of this infrastructure, and they are in use in many higher level NLP tasks, such as rule-based (Trosterud, 2004) and neural disambiguation (Ens et al., 2019), dependency parsing (Antonsen et al., 2010) and

machine translation (Pirinen et al., 2017). The transducers are also in constant use in several real world applications such as online dictionaries (Rueter and Hämäläinen, 2019), spell checkers (Trosterud and Moshagen, 2021), online creative writing tools (Hämäläinen, 2018), automated news generation (Alnajjar et al., 2019), language learning tools (Antonsen and Argese, 2018) and documentation of endangered languages (Gerstenberger et al., 2017; Wilbur, 2018). As an additional important application we can mention the wide use of FSTs in the creation of Universal Dependencies treebanks for low-resource languages, at least with Erzya (Rueter and Tyers, 2018), Northern Saami (Tyers and Sheyanova, 2017) Karelian (Pirinen, 2019a) and Komi-Zyrian (Partanen et al., 2018).

Especially in the context of endangered languages, accuracy is a virtue. Rule-based methods not only serve as NLP tools but also as a way of documenting languages in a machine-readable fashion. Members of language communities do not benefit, for example, from a neural spell checker that works to a degree in a closed test set, but fails miserably in real world usage. On the contrary, a rule based description of morphology can only go so far. New words appear and disappear all the time in a language, and keeping up with that pace is a never ending job. This is where neural models come in as they can learn to generalize rules for out-of-vocabulary words as well. Pirinen (2019b) also showed recently that at least with Finnish the neural models do outperform the rule-based models. This said, Finnish is already a larger language, so the experience doesn't necessarily translate into low-resource scenario (see Hämäläinen 2021).

The purpose of this paper is to propose neural models for the three different tasks the GiellaLT FSTs can handle: morphological analysis (i.e. given a form such as *kissan*, produce the

¹<https://github.com/mikahama/uralicNLP/wiki/Neural-morphology>

²<http://doi.org/10.5281/zenodo.3926769>

³<http://doi.org/10.5281/zenodo.3928628>

morphological reading (+*N+Sg+Gen*), morphological generation (i.e. given a lemma and a morphology, generate the desired form such as *kissa+N+Sg+Gen* to *kissan*) and lemmatization (i.e. given a form, produce the lemma such as *kissan* to *kissa* ‘a cat’). The goal is not to replace the FSTs, but to produce neural fallback models that can be used for words an FST does not cover. This way, the mistakes of the neural models can easily be fixed by fixing the FST, while the overall coverage of the system increases by the fact that a neural model can cover for an FST.

The main goal of this paper is not to propose a state of the art solution in neural morphology. The goal is to first build the resources needed to train such neural models so that they will follow the same morphological tags as the GiellaLT FSTs, and secondly train models that can be used together with the FSTs. All of the trained models will be made publicly available in a Python library that supports the use of the neural models and the FSTs simultaneously. The dataset built in this paper and the exact train, validation and test splits used in this paper have been made publicly available for others to use on the permanent archiving platform Zenodo.

2 Constructing the Dataset

We are well aware of the existence of the popular UniMorph dataset (McCarthy et al., 2020). However, it does not suit our needs of two reasons. One reason is the incompatible morphological tagset. Our goal is to build models that can directly be used side-by-side with the existing FSTs, which means that the data has to follow the same formalism. Conversion is not a possibility, as the main reason we are not interested in using the UniMorph data is its limited scope; not only does it not cover all the languages we are dealing with in this paper, but it does not cover any cases of complex morphology. For example, the Finnish dataset does not cover possessive suffixes, question markers, comparative, superlative etc. Such a data would not be on par with the output produced by the FSTs.

We produce the data for the following languages: German (*deu*), Kven (*fkv*), Komi-Zyrian (*kpv*), Mokhsa (*mdf*), Mansi (*mns*), Erzya (*myv*), Norwegian Bokmål (*nob*), Russian (*rus*), South Sami (*sma*), Lule Sami (*smj*), Skolt Sami (*sms*), Võro (*vro*), Finnish (*fin*), Komi-Permyak (*koi*),

Latvian (*lav*), Eastern Mari (*mhr*), Western Mari (*mrj*), Namonuito (*nmt*), Olonets-Karelian (*olo*), Pite Sami (*sje*), Northern Sami (*sme*), Inari Sami (*smn*) and Udmurt (*udm*). A vast majority of these languages are greatly endangered (Moseley, 2010).

We use the FSTs and dictionaries from the GiellaLT with the UralicNLP (Hämäläinen, 2019) library to build the datasets for training the models. We do this in a clever way by taking all open class part-of-speech words from the dictionaries for each language and use the FSTs to produce all morphological readings for them. The number of words in the GiellaLT dictionaries is shown in Table 1. The FSTs do not let us do this by default, so we build a regular expression transducer that finds all possibilities for an input word and its part-of-speech. In order to build the regular expression, we query all alphabets in the transducer that contain one of the following strings for exclusion: *#*, *Der*, *Cmp* or *Err*. This will remove compounds, erroneously spelled forms and derivations. Derivations need to be excluded because otherwise the transducers would produce derivations of derivations of derivations and so on. Once the regular expression transducer is composed with the FST analyzer, we can use HFST to extract the transducer paths to get a list of all the possible morphological forms of the input word. From these, we filter out *Clt* and *Foc* tags because these multiply the number of possible morphological forms, especially since multiple different clitics can be appended after each other, and some times even in multiple different orders. We also remove tags indicating non-standard forms, *Use* and *Dial*, and *Sem* tags that are used in language learning tools as well as contextual disambiguation to categorize semantically similar words. Table 2 shows how many unique inflectional forms each part-of-speech category has per language.

We use the method described above to produce the data with all the open class part-of-speech words in the GiellaLT dictionaries for each language. For languages with bigger dictionaries, the maximum number of lemmas used per part of speech is set to 2100, in which case the lemmas are also picked at random. We use the typical split ratio and split 70% of the data for training, 15% for validation and 15% for testing. The split is done on the lemma level and for each part-of-speech separately. This means that the test and valida-

	deu	fin	fkv	koi	kpv	lav	mdf	mhr	mns	mrj	myv	nob	olo	rus	sjc	sma	sme	smj	smn	sms	udm	vro
N	8741	51916	5936	558	20042	9738	17196	14079	2263	2529	10234	32009	5942	24691	2685	5946	37943	4331	13826	21158	10722	4703
Adv	588	6036	652	89	2942	953	1771	2346	-	444	743	1743	14	2546	-	543	1314	343	1146	1729	985	122
V	4021	27875	1445	532	12504	2601	11983	9954	4924	2456	3781	7432	2782	14348	1751	5208	7724	3130	5436	5033	3669	4129
A	2768	13056	917	128	5218	1652	4407	5116	-	1031	2926	3236	2134	11054	185	645	2927	468	2295	3898	1550	1019

Table 1: The sizes of the GiellaLT dictionaries per part-of-speech

	deu	fin	fkv	koi	kpv	lav	mdf	mhr	mns	mrj	myv	nob	olo	rus	sjc	sma	sme	smj	smn	sms	udm	vro
N	24	850	50	788	183	24	83	208	151	162	19	17	98	75	16	50	727	297	496	339	744	26
Adv	1	16	1	2	4	-	4	3	-	2	2	2	-	1	-	3	8	3	2	3	1	6
V	254	6667	139	198	249	1245	894	59	-	40	10	21	726	693	38	58	302	144	382	177	156	119
A	150	1244	77	4	244	44	127	4	-	2	5	15	217	39	52	75	1347	187	100	627	54	100

Table 2: Number of unique inflectional forms per part-of-speech category

tion sets will consist exclusively of out of vocabulary words that have not appeared in the training in any inflectional form. This also means that the ratios are the same for each part-of-speech, 70% of the adjectives are used in the training, 70% of the verbs and so on. The actual sizes can be seen in Table 3.

The reason why we do the testing purely on out-of-vocabulary words is simply to test the accuracy of the models in the scenario that is more close to the one they are trained for, namely, in cases where the FSTs fail in their coverage.

3 Experiments and Results

In this section, we cover the neural architecture for the three separate morphological tasks: lemmatization, analysis and generation. We also show the results of the models in these tasks for each language, and present an error analysis on the Finnish and Komi-Zyrian by taking a closer look at the results.

3.1 The Neural Model

Over recent years, there has been a growing body of work on different neural approaches for low resourced languages in morphological analysis (Moeller et al., 2019; Schwartz et al., 2019), lemmatization (Kondratyuk, 2019; Silfverberg and Tyers, 2019) and generation (Oseki et al., 2019; Yu et al., 2020). Most notably the use of bi-directional LSTM architecture seems to be supported by most of the recent related work for analysis, generation and lemmatization.

It is important to note that we approach the lemmatization and analysis from the same point of view as the FSTs. This means that it is a strictly morphological process, and the question of disambiguation is left for another part of the GiellaLT NLP pipeline, namely constraint grammar rules

(Bick and Didriksen, 2015). There is a plethora of work dealing with in-context lemmatization (Manjavacas et al., 2019; Malaviya et al., 2019), morphological analysis (Lim et al., 2018; Zalmout and Habash, 2020) and part-of-speech tagging (Perl et al., 2020; Hoya Quecedo et al., 2020), but that is not what we are aiming for. We are aiming for neural models that can be used to complement the already existing systems relying on the GiellaLT infrastructure.

For all three tasks, we train a character based bi-directional LSTM model (Hochreiter and Schmidhuber, 1997) by using OpenNMT-py (Klein et al., 2017) with the default settings except for the encoder where we use a BRNN (bi-directional recurrent neural network) (Schuster and Paliwal, 1997) instead of the default RNN (recurrent neural network) as BRNN has been shown to provide a performance gain in a variety of tasks. We use the default of two layers for both the encoder and the decoder and the default attention model, which is the general global attention presented by Luong et al. (Luong et al., 2015).

Table 4 shows an example of the input and output of the training data in each of the three different tasks. Words are split into characters on both the input and output side of the data. Different morphological tags are treated as separate tokens, this means that FST morphologies consisting of multiple tags such as *N+Msc+Sg+Dat* are simply split by the plus sign. We train a separate model for each task, meaning that we train three different models for each language: one for lemmatization, analysis and generation. All models have shared the same random seed (3435), therefore training the models again with this seed should result in the exact same results we are reporting in this paper.

	deu	fin	fkv	koi	kpj	lav	mdf	mhr	mns	mrj	myv	nob	olo	rus	sjc	sma	sme	smj	smn	sms	udm	vro
train	394k	14486k	286k	483k	873k	320k	1267k	666k	283k	232k	45k	37k	1054k	243k	80k	108k	799k	648k	1167k	2831k	943k	257k
val	87k	3061k	62k	105k	186k	68k	276k	142k	60k	50k	9k	8k	229k	51k	17k	22k	177k	145k	249k	628k	202k	54k
test	84k	3109k	60k	105k	186k	68k	274k	142k	60k	50k	9k	8k	221k	53k	16k	23k	179k	143k	253k	624k	203k	55k

Table 3: Sizes of the datasets for each language. The splits do not share vocabulary.

	input	output
lemmatization	k a u n i i m p a n s a k o	k a u n i s
analysis	k a u n i i m p a n s a k o	A Comp Sg Gen PxSg3 Qst
generation	k a u n i s A Comp Sg Gen PxSg3 Qst	k a u n i i m p a n s a k o

Table 4: Example of the training data for each task

3.2 Results

We report the performance of the models in terms of accuracy, meaning how many results were fully right (entirely correct lemma, entirely correctly generated form and entirely correct morphological analysis). In addition, we report CER (character error rate) for the lemmatizers and generators, and a MER (morphological error rate) for the analyzers. These values indicate how close the model got to the correct result even if some of the results were a bit erroneous.

The results can be seen in Table 5, the models reaching to an accuracy to over 80 % are highlighted in bold. The results indicate that lemmatization is the easiest task for the model to learn, and after that generation. Morphological analysis is the most difficult task as it receives the scores lower than the generation or lemmatization. Needless to say, some results are exceptionally good for specific languages such as for Erzya (myv) and Western Mari (mrj), while they are not good for others like Finnish (fin) and German (deu). This calls for more investigation of the results.

Figure 1 shows the accuracy of each model based on the morphological complexity of the input. The complexity is measured by the number of morphological tags in the FST produced data. The complexity axis of the plots shows a relative complexity for each language, meaning that 1.0 has the maximum number of tags, 0.8 shows results for input having 80% of the maximum number of tags and so on. The maximum complexity is shown in brackets after the language ISO-code. Analyzers seem to have a lower accuracy for most of the languages when the complexity is small. This is probably due to the fact that shorter word forms tend to have more ambiguity to begin with and might be analyzed as a word different from the one in the gold standard. For many languages, the accuracy

increases towards the average complexity and drop again for the most complex forms. It is to be remembered that these accuracies are also affected by the peculiarities of the transducers themselves and their tagging conventions.

Lemmatizers seem to follow the pattern of the analyzers but do so more clearly. Lemmatization of morphologically simple forms is not as easy as more complex forms. However, as the complexity increases, the lemmatization accuracy does not drop for most of the languages. This has probably something to do with the fact that unlike morphological tags, the word forms follow clearer patterns as they do not have such a large amount of subjectivity in the tagging decisions the different linguists working on these transducers have introduced.

Generators are very even for most of the languages in the sense that they produce consistently around the same accuracy regardless of the morphological complexity. Although, some of the languages follow a more analyzer like pattern, generating wrong with small and large morphological complexity.

Table 6 shows the most difficult tags for the analyzers. The missing predictions column shows the most frequent tags the analyzer did not predict even though they were in the gold data, and the wrong predictions column shows the most frequent ones the analyzer predicted but were not in the gold data. We can see that many of the most challenging tags are shared by different languages. In various Uralic languages, for example, connegatives and imperatives, or connegatives and infinitives, are homonymous, and cannot be predicted correctly just from the surface form alone. Similarly cases such as illative and inessive are in many complex forms homonymous in Permic languages, which surfaces in missing pre-

	deu	fin	fkv	koi	kpv	lav	mdf	mhr	mns	mrj	myv	nob	olo	rus	sje	sma	sme	smj	smn	sms	udm	vro
gen acc	0,65	0,64	0,68	0,67	0,78	0,95	0,85	0,58	0,78	0,90	0,93	0,94	0,83	0,97	0,77	0,69	0,73	0,67	0,57	0,40	0,87	0,82
gen CER	5,61	8,03	3,70	8,67	3,75	1,21	1,77	11,76	3,92	1,77	0,67	1,23	2,12	0,48	4,28	5,81	4,19	3,54	4,25	6,65	1,90	3,35
lem acc	0,88	0,68	0,80	0,70	0,87	0,85	0,93	0,88	0,79	0,88	0,90	0,76	0,87	0,82	0,72	0,71	0,06	0,70	0,67	0,79	0,92	0,79
lem CER	2,71	12,37	5,85	11,41	1,21	4,34	1,11	2,46	4,87	3,82	1,50	5,71	3,76	4,25	6,78	6,96	55,72	9,13	7,78	4,45	2,75	5,81
ana acc	0,11	0,57	0,86	0,78	0,88	0,39	0,61	0,94	0,77	0,92	0,98	0,49	0,86	0,36	0,73	0,60	0,56	0,53	0,42	0,42	0,76	0,74
ana MER	35,40	16,66	6,52	7,24	3,06	18,24	11,54	3,73	7,76	4,75	0,41	38,09	5,85	19,04	19,45	22,91	17,24	22,48	23,20	20,11	10,90	9,82

Table 5: Results of the models for different languages on out-of-vocabulary data

	Missing predictions	Wrong predictions
deu	Def, Pl, Acc, Dat, Neu, Gen, Msc, Fem, NoArt, Nom	NoArt, Fem, Msc, Indef, Sg, Gen, Acc, Nom, Def, Neu
fin	PxSg3, A, PxPl3, N, Sg, Pl, Nom, Gen, Par, Pss	V, Act, PxPl3, PrfPrc, PxSg3, Ind, PrsPrc, Pss, Prs, Sg
fkv	A, Act, N, Sg, V, Pl, Ind, Inf3, Nom, Pl3	N, Act, A, V, Sg, Pl, Ind, Pass, Prs, Inf3
koi	IV, TV, AprIne, AprIll, Ill, Prs, V, So/CP, Apr, Ind	Apr, So/CP, TV, Ine, AprIne, Fut, Sg, N, IV, Nom
kpv	Ine, TV, IV, Fut, Prs, Ill, V, Pl3, Sg1, PxSg1	Ill, IV, TV, Prs, Ine, Fut, Sg, Sg3, N, Pl1
lav	IV, Fem, Acc, Pl, Sg, Nom, TV, Voc, Def, Gen	TV, Gen, Msc, Sg, Pl, Indef, Loc, IV, Fem, Acc
mdf	IV, TV, V, Ind, Prt2, OcPl3, N, A, PxSg2, NomAct	TV, IV, N, Conj, OcSg3, Def, A, V, OcPl1, ScSg1
mhr	N, Sg, V, So/CP, Nom, Ill, Ind, So/PC, Gen, Ger	V, Ind, N, Sg, Adv, Nom, Prs, Sg3, Imprt, A
mns	PxPl2, Sg, Pl, PxDu2, Nom, PxSg2, Du, Lat, Abl, Loc	PxDu2, Pl, Sg, PxSg2, Nom, PxPl3, Du, Lat, Tra, PxPl2
mrj	Sg, N, Lat, Prs, V, Nom, Ind, Imprt, Ine, PxPl3	Prt1, Ind, V, Ill, N, Sg, Nom, Sg3, Ine, Prs
myv	N, A, Tra, Ela, Abl, Ine, Interr, PxSg2	A, N, Abl, Ine, Ela, Ill, Tra, NomAg, V, IV
nob	A, Pos, Indef, Sg, Pl, PrfPrc, Def, Fem, V, MF	Msc, Ind, V, Sg, Indef, Prt, Neu, Def, N, Pl
olo	Ins, N, A, ConNeg, V, Sg, Act, PrfPrc, Nom, Pl	Gen, ConNeg, Act, V, N, Ind, A, Sg, Prs, PrsPrc
rus	TV, Acc, Neu, Gen, Anim, Inan, Impf, Dat, Pass, IV	IV, Loc, Msc, AnIn, Nom, Perf, TV, Acc, Gen, Sg
sje	Sg, V, Com, Prs, N, Sg2, Pl, Gen, Ind, Nom	N, Sg, Pl, V, Ind, Nom, Prs, Prt, Gen, Ine
sma	IV, TV, N, Ind, V, Ess, Sg, Pl, A, Prs	Sg, IV, TV, N, Ind, Com, Prs, Pl, Nom, Ill
sme	Acc, Gen, TV, Sg, Pl, Loc, IV, N, Com, Nom	Gen, Sg, Acc, IV, TV, Nom, Pl, Com, Loc, A
smj	Com, Sg, N, Pl, PxDu2, PxDu1, Gen, Acc, IV, NomAg	Gen, TV, V, PxPl2, PxPl1, Pl, Sg, Nom, IV, Ine
smn	Acc, Sg, Gen, PxPl2, PxSg3, N, PxPl3, IV, PxPl1, TV	PxDu3, Gen, PxDu2, Nom, A, Ill, Sg, PxSg1, PxSg2, Pl
sms	N, Sg, V, Acc, Pl, Gen, Ill, Com, Ind, Nom	A, Pl, Nom, Sg, Gen, Loc, Acc, Com, Ill, Par
udm	Ill, N, Opt, Sg, ConNeg, Ind, PxSg3, Imprt, Sg2, Ine	ConNeg, Ind, Ine, Sg3, Fut, Nom, N, A, Det, Sg
vro	A, Pss, Act, V, Pl, Sg, Ind, N, Gen, ConNegII	Sg, Nom, Act, N, A, Pl, Sg1, Prs, Ind, Par

Table 6: The top 10 most difficult tags for the analyzers

dictions of all these languages. In the languages where transitivity is a feature coded into FST, there are regular problems in predicting these categories correctly. Similarly, in many Indo-European languages gender is primarily a lexical category, and in many instances the model cannot predict it correctly in cases where only the surface form that doesn't show the gender is presented. In the Section 3.3 we go through more in detail this kind of instances, for example, in relation to purely lexically determined Komi-Zyrian stem consonants.

Table 7 shows the morphological constructions that were the most difficult ones for the models to lemmatize and generate correctly in their respective columns. For instance, the Erzya (myv) generation indicates the translative with subsequent possessive-suffix marking is the most problematic. If it had been lemmatization, the explanation would point to the extreme infrequency of these translative forms and the fact that there is an ambiguity with genitive and nominative forms of derivations in *ks*. Lemmatization for Erzya, however, appears to have no issues with ambiguity at all. The same difficulties are not shared by

other languages, but seem to all be language specific. Eastern and Meadow Mari (mhr), for example, appear to have difficulties with generation and lemmatization of nearly the same tag set, namely, the illative plural with a third person plural possessive suffix (ordered: possessive, plural and finally case marker). Looking at the sibling language Western Mari (mrj), we will note that there is a different tagging strategy in use, but here as well there seems to be an intersection where the same forms present problems for both generation and lemmatization.

This could be seen as a type of sanity test whereby simple flaws in the transducers might be detected. The Latvian (lav) transducer is a blatant example of inconsistencies in transducer development. The problem, which has now been addressed and corrected, was in the multiple exponence of part-of-speech tags, i.e. there are double +V and +N tags due to the introduction of automated part-of-speech tagging in XML dictionary to FST formalism transformation without removing the part-of-speech tagging in subsequent continuation lexica of the rule-based transducer.

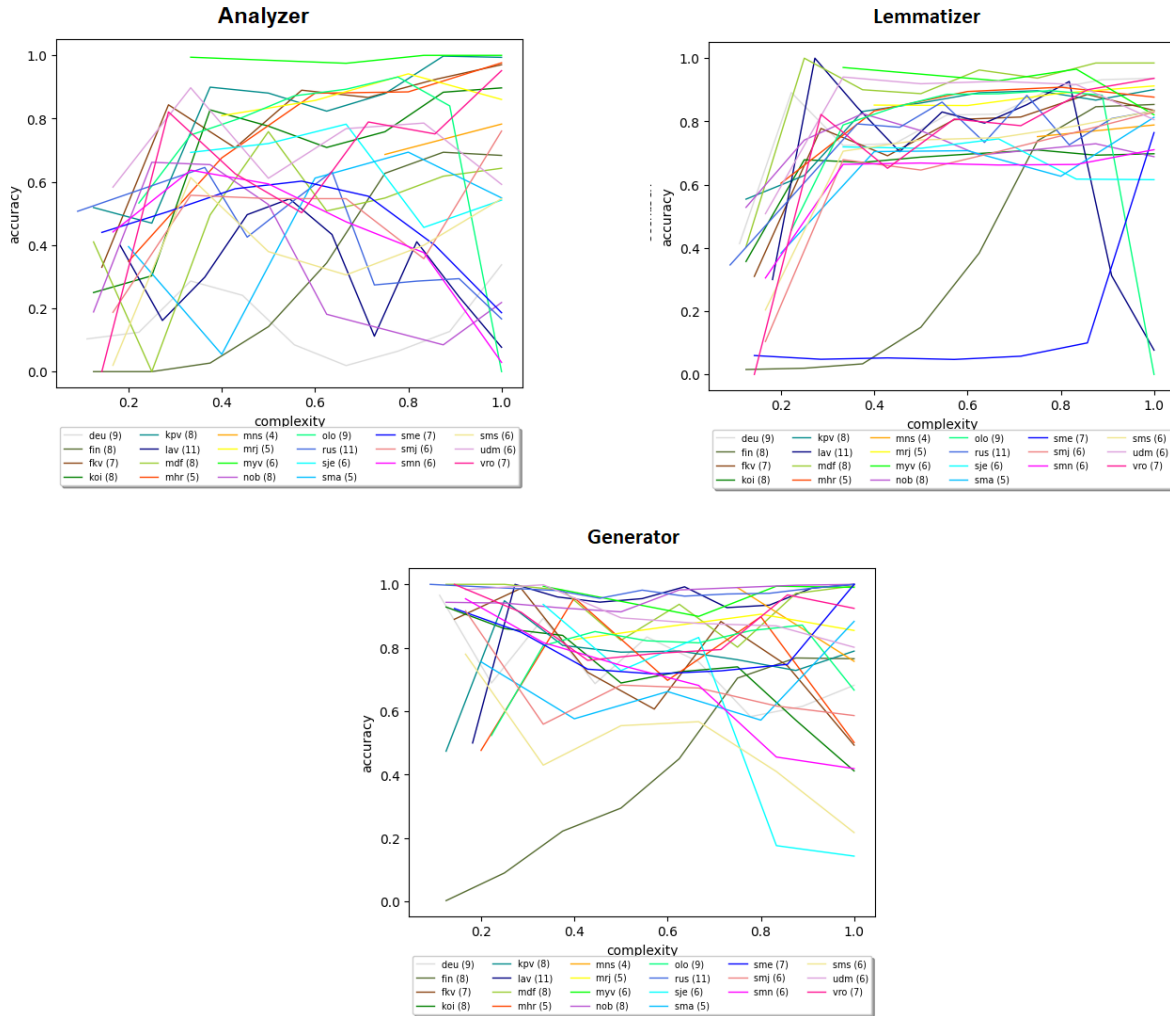


Figure 1: Accuracies based on morphological complexity

Development of the Mari pair might be greatly enhanced through the introduction of a segment-ordering tag in Western or Hill Mari (mrj), which would bring it closer to the strategy followed in the Eastern and Meadow Mari (mhr) use of *+So/PNC*. These questions with tag and suffix ordering appear also as important factor in Komi-Zyrian morphological generation, as discussed in Section 3.3.

3.3 Error Analysis

In this section, we take a closer look at the result of the Finnish (fin) and Komi-Zyrian (kpv) models in order to better understand their shortcomings.

3.3.1 Finnish

For lemmatization Finnish offered one of the worst results, which makes it an interesting target for error analysis. Some of the obvious errors are re-

lated to extremely common word formation patterns, which the model for some reason is not able to generalize. One of these pattern belongs to adjectives and nouns formed with suffix *-inen*, for example *pienimuotoisissani* ‘in my most minor (things)’ the correct lemmatization would be *pienimuotoinen*, but the model returns *pienimuotoida*, which doesn’t mean anything. Interestingly, it gives very consistently similar forms to different variants of the same word, so the model appears to believe this is the correct lemma. We can analyze that out of all Finnish lemmatization errors *-inen* derivations are involved in 7.7% of all mistakes. Thereby future work should investigate what can cause such a gap in the models prediction abilities, as impact in this can lead into rapid improvements. One phenomena we observed is that Finnish FST also produces incorrect forms, such as *pienimuo-*

	generator	lemmatizer
deu	V+PrfPrc+Pos+Pl+Nom+Indef, V+PrfPrc+Pos+Fem+Sg+Nom+Indef, V+PrfPrc+Pos+Pred, V+PrfPrc+Pos+Fem+Sg+Acc+Indef, V+PrfPrc+Pos+Neu+Sg+Acc+Def	N+Msc+Pl+Dat, N+Msc+Pl+Gen, N+Msc+Pl+Nom, N+Msc+Pl+Acc, N+Msc+Sg+Dat
fin	A+Sg+Ess+PxSg3, A+Sg+Ess+PxPl3, A+Sg+Ess+PxPl3+Qst, A+Sg+Ess+PxSg3+Qst, N+Pl+Par+PxPl3+Qst	A+Sg+Ess+PxSg3, A+Sg+Ess+PxPl3, A+Sg+Ess+PxPl3+Qst, A+Sg+Ess+PxSg3+Qst, N+Pl+Par+PxSg3
fkv	V+Act+Inf3+A+Pl+Superl+Par, A+Pl+Superl+Par, N+Pl+All, V+Act+Inf3+A+Pl+Par, V+Act+Inf3+A+Pl+Gen	N+Pl+All, N+Pl+Par, N+Pl+Gen, N+Sg+Par, N+Pl+Abe
koi	V+Ind+Prt2+Pl3+Comp, V+Ind+Prt2+Pl3, V+IV+Ind+Prt2+Pl3+Comp, V+IV+Ind+Prt2+Pl3, V+TV+Ind+Prt2+Pl3	N+Sg+Ela+Comp+Cop+Pl, N+Sg+Ine+PxPl1+Comp+Cop+Pl, N+Sg+Ine+PxPl2+Comp+Cop+Pl, N+Sg+Ine+PxPl3+Comp+Cop+Pl, N+Sg+Ela+PxSg1+Comp+Cop+Pl
kpv	N+Sg+Com+PxSg2, N+Sg+Com+PxSg3, N+Sg+Egr+PxSg1+Comp, N+Sg+Egr+PxSg1, N+Sg+Egr+PxSg1+Comp+Cop+Pl	N+Sg+Acc, Adv, N+Sg+Prt+PxPl1, N+Sg+Com+PxSg2, N+Sg+Com+PxSg3
lav	V+V+TV+PrsPrc+Act+Msc+Sg+Voc+Def, V+V+TV+PrsPrc+Pss+Msc+Sg+Voc+Def, V+V+TV+PrfPrc+Pss+Msc+Sg+Voc+Def, N+N+Msc+Sg+Voc, V+V+IV+PrsPrc+Act+Msc+Sg+Voc+Def	N+N+Msc+Sg+Voc, N+N+Fem+Sg+Voc, N+N+Fem+Pl+Gen, N+N+Fem+Sg+Acc, N+N+Fem+Sg+Loc
mdf	V+IV+NomAg+Pl+Gen+PxSg3, V+IV+NomAg+Pl+Nom+PxSg3, V+IV+NomAg+SP+Cau+Indef, V+IV+NomAg+Sg+Dat+PxSg1, V+IV+NomAg+Pl+Dat+PxSg3	N+Sg+Nom+Indef, N+SP+Gen+Indef, N+Sg+Dat+PxSg2, N+Sg+Dat+PxSg1, N+SP+Tra+Indef
mhr	N+Pl+III+PxSg3+So/PNC, N+Pl+III+PxPl1+So/PNC, N+Pl+III+PxPl2+So/PNC, N+Pl+III+PxPl3+So/PNC, N+Pl+III+PxSg1+So/PNC	Adv, N+Pl+III+PxSg3+So/PNC, N+Pl+III+PxSg2+So/PNC, N+Sg+III+PxSg3+So/CP, N+Pl+III+PxSg1+So/PNC
mms	N+Du+PxDu1+Abl, N+Du+PxDu1+Ins, N+Du+PxDu1+Nom, N+Du+PxDu1+Loc, N+Du+PxDu1+Lat	N+Pl+PxSg2+Ins, N+Pl+PxSg2+Loc, N+Pl+PxSg2+Abl, N+Pl+PxSg2+Nom, N+Pl+PxSg2+Lat
mrj	N+Sg+Ine+PxSg3, N+Sg+III, N+PxSg2+Pl+III, N+Sg+PxSg2+III, N+PxSg1+Pl+Lat	N+Sg+III, N+Sg+Gen, N+Sg+Acc, N+Sg+Nom, N+Sg+Ine+PxSg3
myv	N+SP+Tra+PxSg2, N+SP+Tra+PxPl1, N+SP+Tra+PxPl2, N+SP+Tra+PxPl3, N+SP+Tra+PxSg3	V+IV+Act+PrsPrc, V+IV+NomAg+SP+III+PxSg2, V+IV+NomAg+SP+III+PxPl1, V+IV+NomAg+SP+III+PxSg3, V+IV+NomAg+SP+Ine+PxPl1
nob	N+Neu+Pl+Def, V+Ind+Prt, N+Neu+Pl+Indef, V+PrfPrc, A+Superl+Def	V+Imp, A+Pos+Neu+Sg+Indef, A+Pos+Fem+Sg+Indef, V+Ind+Prt, A+Pos+Msc+Sg+Indef
olo	V+Act+PrsPrc+Pl+Abe, V+Act+PrsPrc+Pl+Abe+Qst, N+Pl+Abe, N+Pl+Abe+Qst, N+Sg+Abe+Qst	N+Sg+Nom, N+Sg+Nom+Qst, N+Sg+Abe, N+Pl+Abe+Qst, N+Sg+Abe+Qst
rus	V+Perf+IV+Imp+Pl2, V+Perf+IV+Imp+Sg2, V+Perf+IV+Fut+Sg3, V+Perf+IV+Fut+Sg2, V+Perf+IV+Fut+Sg1	Adv, A+Neu+Sg+Pred, A+Msc+Sg+Pred, V+Perf+IV+Imp+Sg2, A+Msc+AnIn+Sg+Loc
sje	N+Pl+Ela, N+Pl+Com, N+Sg+Ela, N+Sg+Com, V+Pot+Sg3	N+Pl+Com, N+Pl+Ela, N+Sg+Ela, N+Sg+Com, V+Ind+Prs+Sg3
sma	N+Pl+Gen, N+Pl+Com, N+Sg+Com, N+Pl+III, N+Ess	N+Pl+Gen, N+Sg+Gen, N+Ess, N+Pl+Ine, N+Pl+Nom
sme	A+Comp+Sg+Nom+Qst, A+Comp+Sg+Nom, A+Comp+Attr, A+Comp+Attr+Qst, V+TV+VAbess+Qst	A+Comp+Sg+Nom+Qst, A+Comp+Sg+Nom, A+Comp+Attr, A+Comp+Attr+Qst, V+TV+VAbess
smj	N+Sg+Com+PxSg1, N+Sg+Com+PxSg2, N+Sg+Abe, N+Pl+Abe, N+Pl+Gen+PxSg1	N+Pl+Abe, N+Sg+Abe, N+Sg+Com+PxSg1, N+Sg+Com+PxSg2, N+Pl+Com+PxSg2
smn	N+Pl+Com+Qst, A+Pl+Com+Qst, A+Comp+Pl+Com+Qst, A+Superl+Pl+Com+Qst, V+PrsPrc+Qst	N+Pl+Com+Qst, N+Pl+Gen+Qst, V+Ind+Prs+Sg3+Qst, A+Pl+Com+Qst, V+Ind+Prs+ConNeg+Qst
sms	A+Superl+Sg+Abe, A+Superl+Sg+Abe+Qst/a, A+Superl+Sg+Abe+Qst/ko, V+VAbess+Qst/a, V+VAbess+Qst/ko	V+Ind+Prt+Pl1, V+VAbess+Qst/a, V+Ind+Prt+Pl1+Qst/ko, V+VAbess+Qst/ko, V+VAbess
udm	N+Sg+Ela+PxPl1, N+Sg+Ela+PxSg3, N+Sg+Ela+PxSg2, N+Sg+Ela+PxPl3+Qst, N+Sg+Ela+PxSg1	V+Ind+Prs+Pl1, V+Ind+Prs+Pl1+Qst, V+Ind+Fut+Pl1+Qst, V+Ind+Fut+Pl1, V+Imp+Pl2
vro	V+Act+Sup+Ine, V+Act+Ind+Prt+Sg2, V+Pss+Ind+Prt+Sg2, V+Pss+PrfPrc, V+Pss+PrfPrc+Sg+Nom	V+Act+Ind+Prt+Sg2, N+Pl+III, V+Pss+PrfPrc, V+Pss+PrfPrc+Sg+Nom, V+Pss+Ind+Prt+Sg2

Table 7: The top 5 morphological forms that were the most difficult to lemmatize and generate

toisimmilleän, which probably should end into *-een*. We can also observe that in many Finnish lemmas that the model does analyze correctly the forms are compounds. This leaves open the possibility that the training data has contained either the second component independently or within a comparable compound, which would have given the model some example. One lemmatization issue that can be distinguished is that the model doesn't lemmatize correctly proper names that are written with initial capital letter. These include several words, for example *Unkareinansako* 'as their Hungarians?' should be lemmatized as *Hungary*, but the model returns *nkareintaa*. What this shows is that the model struggles with uppercase characters, although those would ideally be part of the correct lemmatization result.

The Finnish model has problems in generating forms for words ending in *-lainen*, as it seems to inflect them as one would inflect the word *laine* 'wave', such as *dominikaanilaineiltasi* '≈ from

your Dominican waves' instead of *dominikaanilaisiltasi* 'from your Dominican people'. Also, other adjectives ending in *-inen* are problematic such as *keväneensä* instead of *keväisensä* 'his spring-like'. In this case, the model has not learned the typical inflectional category of adjectives ending in *-inen*. This issue has an interesting parallel with the same problem being present in the lemmatization task, described above. This shows that the problems the models encounter are to some degree parallel to one another in different tasks, and either relate to the complexity of the linguistic system, or somehow inadequately represented input.

Interestingly, the generation model has problems with the plural forms of the abessive and illative case, and often generates the singular form instead of the plural such as in *sähkömittarikesi* 'for your electricity meter' instead of *sähkömittareiksesi* 'for your electricity meters' or a completely erroneous form such as *sähkömittaritsiisi* instead

of *sähkömittareihisi* ‘to your electricity meters’. In these erroneous cases, the model has tried to pluralize the word, for example *sähkömittarit* is the correct plural form of electricity meters in nominative, but it is no longer correct when inflected in the illative case.

3.3.2 Komi-Zyrian

When we examine the lemmatization task, some particularities are obvious in Komi-Zyrian. For example, many of word forms with interspersed white spaces in them are not lemmatized correctly. We also see that some complex entries borrowed from Russian are challenging to lemmatize, possibly due to their rarity, for example: народно-освободительнойджыкъяснысланьджык ‘more in the direction of their people who are more national-liberational’ would correctly result in народно-освободительной, but the model predicts народнотильной. In this case the hyphen within the compound probably contributes to the rarity of the form itself. Similarly, the model is also struggling when there are words that follow orthographic conventions more typical to Russian than Komi, for example областьсаас would be correctly lemmatized as областьса, but the model predicts областььса. If this reflects the underlying code, model training like this could be very useful for locating erroneously coded transducers. The double soft sign would seem to allude to double exponence in the code. The model also has challenges with rarer orthographical conventions in Komi vocabulary. For example пипуа-кыддзаиньланьсянь ‘from the direction of my aspen and birch grove’ should be пипуа-кыддзаин ‘aspen and birch grove’, but we get пипуа-кыдзаин. These shortcomings, however, are relatively rare in the Zyrian data, and the model learns to lemmatize at high accuracy. Much more so than Finnish, which could be related to more concatenative morphology of Komi where the word boundaries can be easier to detect.

In the case of Komi-Zyrian we can observe that a large portion of wrongly recognized forms results from ambiguity that is inherent to the morphology of this language. For example, it is not possible to distinguish some of the cases, such as the inessive and illative, in all forms where they occur. As the model inevitably returns only one reading, it is clear that the evaluation accuracy cannot be perfect. This finding is consistent with

analogous ambiguity for other forms in the Skolt Sami (sms) model. There appears to be a consistency in what is incorrectly predicted in Skolt Sami. When there is a four-way ambiguity as in the *Sg Gen*, *Sg Acc Sg Nom* and *Pl Nom*, the tag *Sg Gen* is consistently predicted to be *Pl Nom*, leaving the two readings *Sg Acc* and *Sg Nom* out of the dichotomy. Komi models shows similar preferences into specific categories when there are multiple homonymous possibilities.

In the analysis above it was already briefly discussed that some categories are difficult to recognize correctly for Permic languages. Another example like this is seen in the Komi-Zyrian and Komi-Permyak (koi) future tense marking. As these languages have morphologically marked future in the third person alone, every first and second person verb in the present tense also gets a future reading, as both analyses can be seen as correct. One could also argue, however, that if some analysis is not possible to resolve at this level, some of the distinctions could be removed or merged at this level of analysis.

What comes to morphological generation of Komi, the accuracy is rather high. Some of the errors can be connected to the fact that some suffixes can occur in varying orders. For example, with input кольквиж A Sg Egr PxPl1 Comp one could assume the output кольквижнымсяньджык ‘more from the direction of our yellows’, but in this case the model outputs кольквижсяньнымджык. The only difference is, however, in the order of markers for case Egr and possessive suffix PxPl1. The model is actually giving a correct output, but the input doesn’t have all information about the suffix order that the model would need.

There are also instances of word generation where the correct prediction would demand actual lexicographical knowledge, which the model cannot have. For example, Komi displays with some nouns an additional stem consonant. It is not possible to predict from the surface form whether this consonant exists and what it is. So when the model is given input мек N Sg Ins, it doesn’t predict the correct мекйөн ‘with a pelt’, but offers the regular but incorrect form мекөн. This is a good example from construction where rule-formulated linguistic knowledge may be necessary for optimal analysis. It also shows that the model is capable to learn very well the regular structures of the lan-

guage and does predict them with high accuracy.

4 Conclusions

In this paper, we have presented a method for automatically extracting inflectional forms from FST transducers by composing a regular expression transducer for each word with an existing FST transducer. This way, we have been able to gather very large morphological training data for analysis, lemmatization and generation for 22 languages, 17 out of which are endangered and fighting for their survival. We have used this dataset to train neural models for each language. Because the data follows the tags and conventions used in the GiellaLT infrastructure, these neural models can be used directly side by side with the FST transducers in many of the applications that depend on them.

The results look very good for some languages while being a bit more modest for others. Analysis seems to be the hardest problem out of the three, and its training also took the longest time. Despite this, many models reached to an over 80% accuracy in the tasks. This is rather good given that the evaluation was conducted entirely on out-of-vocabulary words.

The accuracies reported in this paper are a somewhat lower than what they could be. This is due to the fact that we ran the evaluation by producing one result only for each input with the neural models and compared that input directly to the one in the test data. As we saw in our analysis, many of the inputs in the test data were ambiguous, which caused the neural model to produce an output that is correct, but not the one in the test data. However, the right way to overcome this problem would be to research how to deal with ambiguity. The neural models we trained can already now produce N best candidates for each input.

It is probable that within those N best candidates, the models actually cater for the ambiguity and produce other results that are correct as well. For instance, the Finnish word *noita*, can be an accusative singular noun meaning ‘witch’ or a partitive of *nuo* meaning ‘them’. Knowing how to maximize the number of forms the neural model produces while minimizing the number of incorrect forms is a question for another paper. Although, some methods could already be used with the models trained in this paper by in-

troducing simple modifications to how the results are predicted (Silfverberg and Tyers, 2019).

Even though we aimed for a real world scale morphological tag complexity by querying all possibilities from the FSTs, there are still a couple of morphological categories we did not tackle for practical reasons. One of them is the use of clitics. The problem with these is that they can be attached to almost any kind of word regardless of its part-of-speech and inflectional form. On top of this, multiple clitics can be added one after another. To give an idea of the scale, with clitics, Finnish has 9425 unique forms for nouns (instead of 850), 216 for adverbs (instead of 16), 14794 for adjectives (instead of 1244) and a whopping 88044 forms for verbs (instead of 6667). This means that clitics need to be solved by taking a different approach than the one we had. One could, for example, introduce some forms with different combinations of clitics here and there in the training data, in which case the question arises on how many forms need to appear with clitics in order for the model to generalize their usage.

Compounds and derivations could not be included because of how the FSTs were implemented. If you ask an FST for compounds and derivations, you will surely get them! Even in such quantities that your computer will run out of RAM and swap memory for the forms of a single word, as there is no limit to how many words can be written together to form a compound or how many times one can derive a new word from another. We people might have our cognitive limits for that, but the FSTs will not⁴. The problem of compounds is probably best to leave for a separate model to solve, as there are already methods out there for predicting word boundaries (Shao et al., 2018; Seeha et al., 2020). The compound splits by such methods could then be fed into the neural models trained in this paper. As for derivations, some of them could be included in the training data, but the question of how many forms are needed would still require further research.

⁴Take, for instance, a look at this derivational Skolt Sami word produced by the FST *Piän’njatöövölltâsttiatemesvuot’tsázvuöötövvstölškuäit’teškuättöölstölstâststööstčättömâs* for *piännai+N+Der/Dimin+Der/N2A+Der/toovvyd+Der/oollyd+Der/jed+V+Der/Caus+Der/Dimin+Der/NomAg+N+Der/Dimin+Der/N2A+Der/teqm+A+Attr+Der/vuott+N+Der/sazh+A+Err/Orth+Attr+Der/vuott+N+Der/toovvyd+Err/Orth+Der/stoollyd+V+Der/shkueqtted+Der/jed+V+Der/Caus+Der/shkueqtted+Der/oollyd+Der/stoollyd+Der/Dimin+V+Der/Dimin+Der/Dimin+V+Der/Dimin+Der/ched+Der/Caus+Der/t+A+Superl+Attr*

References

- Khalid Alnajjar, Leo Leppänen, and Hannu Toivonen. 2019. No time like the present: methods for generating colourful and factual multilingual news headlines. In *Proceedings of the 10th International Conference on Computational Creativity*. Association for Computational Creativity.
- Lene Antonsen and Chiara Argese. 2018. Using authentic texts for grammar exercises for a minority language. In *Proceedings of the 7th workshop on NLP for Computer Assisted Language Learning*, pages 1–9.
- Lene Antonsen, Trond Trosterud, and Linda Wiecheteck. 2010. Reusing grammatical resources for new languages. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*.
- Eckhard Bick and Tino Didriksen. 2015. Cg-3—beyond classical constraint grammar. In *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, pages 31–39.
- Jeff Ens, Mika Härmäläinen, Jack Rueter, and Philippe Pasquier. 2019. Morphosyntactic disambiguation in an endangered language setting. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 345–349.
- Ciprian Gerstenberger, Niko Partanen, and Michael Rießler. 2017. Instant annotations in elan corpora of spoken and written komi, an endangered language of the barents sea region. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 57–66.
- Mika Härmäläinen. 2018. Poem machine—a co-creative nlg web application for poem writing. In *The 11th International Conference on Natural Language Generation Proceedings of the Conference*. The Association for Computational Linguistics.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- José María Hoya Quecedo, Koppatz Maximilian, and Roman Yangarber. 2020. Neural disambiguation of lemma and part of speech in morphologically rich languages. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3573–3582, Marseille, France. European Language Resources Association.
- Mika Härmäläinen. 2019. UralicNLP: An NLP library for Uralic languages. *Journal of Open Source Software*, 4(37):1345.
- Mika Härmäläinen. 2021. Endangered languages are not low-resourced! In Mika Härmäläinen, Niko Partanen, and Khalid Alnajjar, editors, *Multilingual Facilitation*. Rootroo Ltd.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proc. ACL*.
- Dan Kondratyuk. 2019. Cross-lingual lemmatization and morphology tagging with two-stage multilingual BERT fine-tuning. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 12–18, Florence, Italy. Association for Computational Linguistics.
- KyungTae Lim, Niko Partanen, and Thierry Poibeau. 2018. Multilingual dependency parsing for low-resource languages: Case studies on north saami and komi-zyrian. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Krister Lindén, Erik Axelsson, Senka Drobac, Sam Hardwick, Juha Kuokkala, Jyrki Niemi, Tommi A Pirinen, and Miikka Silfverberg. 2013. HFST a system for creating NLP tools. In *International Workshop on Systems and Frameworks for Computational Morphology*, pages 53–71. Springer.
- Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Chaitanya Malaviya, Shijie Wu, and Ryan Cotterell. 2019. A simple joint model for improved contextual neural lemmatization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1517–1528, Minneapolis, Minnesota. Association for Computational Linguistics.
- Enrique Manjavacas, Ákos Kádár, and Mike Kestemont. 2019. Improving lemmatization of non-standard languages with joint learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1493–1503, Minneapolis, Minnesota. Association for Computational Linguistics.
- Arya D McCarthy, Christo Kirov, Matteo Grella, Amrit Nidhi, Patrick Xia, Kyle Gorman, Ekaterina Vylomova, Sabrina J Mielke, Garrett Nicolai, Miikka Silfverberg, et al. 2020. Unimorph 3.0: Universal morphology. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 3922–3931.
- Sarah Moeller, Ghazaleh Kazeminejad, Andrew Cowell, and Mans Hulden. 2019. Improving low-resource morphological learning with intermediate forms from finite state transducers. In *Proceedings of the 3rd Workshop on the Use of Computational*

- Methods in the Study of Endangered Languages Volume 1 (Papers)*, pages 81–86, Honolulu. Association for Computational Linguistics.
- Christopher Moseley, editor. 2010. *Atlas of the World's Languages in Danger*, 3rd edition. UNESCO Publishing. Online version: <http://www.unesco.org/languages-atlas/>.
- Sjur Moshagen, Jack Rueter, Tommi Pirinen, Trond Trosterud, and Francis M. Tyers. 2014. Open-Source Infrastructures for Collaborative Work on Under-Resourced Languages. The LREC 2014 Workshop “CCURL 2014 - Collaboration and Computing for Under-Resourced Languages in the Linked Open Data Era”.
- Yohei Oseki, Yasutada Sudo, Hiromu Sakai, and Alec Marantz. 2019. Inverting and modeling morphological inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 170–177, Florence, Italy. Association for Computational Linguistics.
- Niko Partanen, Rogier Blokland, KyungTae Lim, Thierry Poibeau, and Michael Riebler. 2018. The first komi-zyrian universal dependencies treebanks. In *Second Workshop on Universal Dependencies (UDW 2018), November 2018, Brussels, Belgium*, pages 126–132.
- Tal Perl, Sriram Chaudhury, and Raja Giryes. 2020. Low resource sequence tagging using sentence reconstruction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2692–2698, Online. Association for Computational Linguistics.
- Tommi A Pirinen. 2019a. Building minority dependency treebanks, dictionaries and computational grammars at the same time—an experiment in karelian treebanking. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 132–136.
- Tommi A Pirinen. 2019b. Neural and rule-based finnish nlp models—expectations, experiments and experiences. In *Proceedings of the Fifth International Workshop on Computational Linguistics for Uralic Languages*, pages 104–114.
- Tommi A Pirinen, Francis Tyers, Trond Trosterud, Ryan Johnson, Kevin Unhammer, and Tiina Puolakainen. 2017. North-sámi to finnish rule-based machine translation system. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 115–122.
- Jack Rueter and Mika Hämmäläinen. 2019. On xml-mediawiki resources, endangered languages and tei compatibility, multilingual dictionaries for endangered languages. *Rachel Edita O. ROXAS President National University (The Philippines)*, page 350.
- Jack Michael Rueter and Francis M Tyers. 2018. Towards an open-source universal-dependency treebank for erzya. In *International Workshop for Computational Linguistics of Uralic Languages*.
- Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681.
- Lane Schwartz, Emily Chen, Benjamin Hunt, and Sylvia L.R. Schreiner. 2019. Bootstrapping a neural morphological analyzer for st. lawrence island yupik from a finite-state transducer. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages Volume 1 (Papers)*, pages 87–96, Honolulu. Association for Computational Linguistics.
- Suteera Seeha, Ivan Bilan, Liliana Mamani Sanchez, Johannes Huber, Michael Matuschek, and Hinrich Schütze. 2020. ThaiLMCut: Unsupervised pretraining for Thai word segmentation. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6947–6957, Marseille, France. European Language Resources Association.
- Yan Shao, Christian Hardmeier, and Joakim Nivre. 2018. Universal word segmentation: Implementation and interpretation. *Transactions of the Association for Computational Linguistics*, 6:421–435.
- Miikka Silfverberg and Francis Tyers. 2019. Data-driven morphological analysis for uralic languages. In *Proceedings of the Fifth International Workshop on Computational Linguistics for Uralic Languages*, pages 1–14.
- Trond Trosterud. 2004. Porting morphological analysis and disambiguation to new languages. In *SALTMIL Workshop at LREC 2004: First Steps in Language Documentation for Minority Languages*, pages 90–92. Citeseer.
- Trond Trosterud and Sjur Moshagen. 2021. Soft on errors? the correcting mechanism of a Skolt Sami speller. In Mika Hämmäläinen, Niko Partanen, and Khalid Alnajjar, editors, *Multilingual Facilitation*, pages 197–207. Rootroo Ltd.
- Francis Tyers and Mariya Sheyanova. 2017. Annotation schemes in north sámi dependency parsing. In *Proceedings of the Third Workshop on Computational Linguistics for Uralic Languages*, pages 66–75.
- Joshua Wilbur. 2018. Extracting inflectional class assignment in pite saami: Nouns, verbs and those pesky adjectives. In *Proceedings of the Fourth International Workshop on Computational Linguistics of Uralic Languages*, pages 154–168, Helsinki, Finland. Association for Computational Linguistics.
- 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 SIG-MORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 70–78, Online. Association for Computational Linguistics.

Nasser Zalmout and Nizar Habash. 2020. Joint diacritization, lemmatization, normalization, and fine-grained morphological tagging. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8297–8307, Online. Association for Computational Linguistics.