A Template Based Approach for Training NMT for Low-Resource Uralic Languages - A Pilot with Finnish

Mika Hämäläinen*
mika.hamalainen@helsinki.fi
Department of Digital Humanities
Finland

Khalid Alnajjar*
khalid.alnajjar@helsinki.fi
Department of Computer Science & HIIT
Finland

ABSTRACT

In this paper, we present a novel way of building parallel data by filling abstract morphosyntactic structures for endangered low-resource languages that exhibit a rich productive morphology. We use Finnish to pilot our approach by limiting the resources we feed into the machine translation model to the level of the resources available for Erzya. We also present a way of automatically mapping abstract morphosyntactic structures of two languages to produce a parallelized set of templates.

CCS CONCEPTS

• **Computing methodologies → Machine translation**; **Neural networks**; **Language resources**; **Natural language generation**.

KEYWORDS

endangered languages, machine translation, synthetic templates

1 INTRODUCTION

Uralic languages exhibit a great richness in their productive morphology. In all of these languages, we count the number of inflectional forms in hundreds for a vast majority of individual words. Such a richness poses great challenges to neural machine translation approaches as the NMT model has to learn to translate word forms it has never encountered before in its training data.

What makes this task an even more challenging one is when we go from the majority languages into the context of endangered low-resource languages. By and large most of the languages in the Uralic language family, with the exception of Hungarian, Finnish

---

*Both authors contributed equally to this research.

and Estonian, are to some extent endangered and struggling for their existence.

Giellatekno [18] has established a long tradition in NLP research for these languages having developed tools such as FST (finite-state transducer) morphology, RBMT (rulebased machine translation) and CG (constraint grammar) disambiguators among others. This strong rulebased tradition has been guiding the NLP research focusing on these languages, but only little has been done to look into alternatives, such as NMT.

Even if non-rulebased methods were applied, the research will suffer from two unavoidable obstacles: the lack of data and the difficulty of evaluating the results as evaluation would require a native-level command of the language in addition to linguistic meta-knowledge about the structure of the language.

In this paper, we are, however, introducing a method for training an NMT model. We do this by generating parallel data automatically for Finnish and English to solve the problem of rich morphology. We limit the resources we use to a similar set as one of the endangered languages, Erzya, would have. This way we can make sure that positive results from this pilot experiment can be transferred to the context of that language as well.

The reason why we use Finnish in this pilot is that firstly, its morphology is no poorer than those of the endangered Uralic languages, and secondly it allows us to conduct evaluation of the results in a much easier fashion. This allows us to try out different approaches and make immediate initial judgments on the results with our native Finnish knowledge. Such a trial and error way that is quite common place in the NLP world would be difficult if directly tried out with a minority language data, due to the lack of suitable evaluators.

2 RELATED WORK

Many contemporary machine translation models rely on the availability of parallel texts. Nevertheless, parallel texts are a scarce resource, in various domains and languages. This poses great challenges, especially in building parallel corpora for low-resource languages to train data hungry models.

Munteanu and Marcu [19] have proposed a method which employs a large non-parallel, yet comparable corpora consisting of e.g. news articles, along with relatively small parallel corpora of a different domain e.g. United Nations corpus. Building on these improves translations in the desired domain, news articles. Their method attempts to map sentences in the comparable corpora by selecting articles of both languages that share the same topic, pair sentences in these articles together and, then, predict whether the two sentences are translations of each other. Despite the gains a machine translation model would get by employing such methods,

it is not feasible to utilize them in some cases, for instance very low resource Uralic languages e.g. Erzya, due to the lack of such comparable corpora.

A way of building a machine translation model for resource-poor languages is to utilize a resource-rich language, related to the low-resource one, as a parent. This means learning some of its characteristics (e.g. syntax and morphology), and transferring them to the low-resource MT model [14, 20, 21, 26]. These approaches do not necessarily use parallel texts of the low-resource language as they rely on the resource-rich language, which could result in a poor coverage of the morphology of the low-resource language.

Researchers have also investigated approaches to constructing parallel texts by crowdsourcing, i.e. asking online workers to translate a given expression to another language [3, 4, 25]. The small number of native speakers makes crowdsourcing a difficult task in the context of low-resource languages. What makes the task even more difficult is the lack of an established standard or even pluricentricity of the language in question. This makes it more difficult to control the quality of the crowdsourced translation tasks even if a substantial number of workers were reached.

Chahuneau et al. [6] have proposed a method for translating English into morphologically rich languages by building a model for predicting inflections of words in the target language. Thereafter, the model is employed in constructing words and phrases –synthetic phrases–, potentially with new inflections to the training data. The constructed synthetic phrases are then used along with a parallel corpus to train an MT model.

3 GENERATING THE PARALLEL DATA

As the parallel Erzya-English corpus that is provided as a part of the Erzya treebank [23] only consists of around 160 sentences, we can rule out the possibility of using any real parallel data for our pilot either. This means that the data has to be generated. As the Uralic languages opt for agglutinative morphology to express a similar notion the more analytical English would split it in multiple different words. For instance, the Finnish *kävellessänikin* (even when I am walking) conveys a lot of information in something that seemingly is only one word. This means that we cannot feed the NMT model just lists of words from a dictionary and hope it to reach to an acceptable translation, but rather we have generate longer structures mapping the Finnish morphology to an English multi-word structure.

In practice, we will develop a parallel corpus consisting of an abstract level morphosyntactic mapping between English and Finnish. The parallel data consists of mappings as we can see in the following example.

| N+Sg+Ine | in a N |
| N+Sg+Gen takana | behind a N |

In other words, words are replaced by a placeholder stating a part-of-speech and morphology unless they belong to a close-ended part-of-speech category such as adpositions, articles and so on, in which case the words are retained as they are in the parallel data.

This parallel corpus is not used directly as it is for the NMT model, but rather it is used to generate parallel data, by replacing the placeholders with real words and their translations and inflecting according to the morphosyntax in the parallel corpus. In doing so,

we will make sure that the NMT model will have seen all of the words in most of their morhpological forms.

A parallel corpus of this kind can be constructed manually for a low-resource language. However, we are interested to see just how many of these parallelized structures are needed to reach to an acceptable translation, and what is the length of these structures. Is unigram to n-gram enough? Do we also need bigram and trigram to n-gram structures? Piloting this study with Finnish makes answering to these questions a little bit easier than working directly with Erzya.

For Finnish, we do not have to build such a corpus by hand, but instead, we can learn the mapping from existing parallel data. This shortcut makes it possible for us to try out different numbers of parallel structures in a much faster way than having to write them by hand.

3.1 Producing the Morphosyntactic Parallel Corpus

We produce automatically the parallel morphosyntactic corpus from the English-Finnish Bible parallel data [17]. We produce syntactic parsing for the English sentences with Spacy [13] and for the Finnish sentences, we use the Finnish-dep-parser [12], which outputs the parsing in the CoNLL-U format, which we parse with UralicNLP [9].

We create a tokenized parallel morphosyntactic corpus from the parsed data. This is done by replacing the words belonging to open class part-of-speech categories (nouns, adjectives, verbs, adverbs) with placeholders containing their part-of-speech together with their morphological analysis. For instance, we replace the occurrences of the Finnish word *koirissa* with *NOUN_Case=Ine _Number=Plur*. We do this for both English and Finnish sentences leaving the closed class words (such as adpositions and pronouns) as they are. As a result, we have parallel sentences from the Bible where some of the words are replaced by these placeholders bearing a more abstract level morphological information.

Thereafter, we iterate over all of these parallel sentences in the data. Our endeavor is to map unigram and bigram tokens in the morphosyntactically parsed Finnish sentences into n-gram English tokens. In the English side, these n-grams can vary from unigrams up to 5-grams. We allow a bigger n-gram window for English, because Finnish has more complex morphology and a word could be described with multiple words in English. Hence, for each Finnish sentence, we calculate the number of times a Finnish unigram or bigram co-occur with an n-gram, where $1 \leq n \leq 5$, English token.

Once all co-occurences are obtained, we need to find the most descriptive and likely mappings between the morphosyntactic structures of the two languages. To achieve that, we opt for the simple log-likelihood association measure Evert [7], using the equation below.

$$ll = 2(O \cdot log \frac{O}{E} - (O - E))$$ (1)

In the equation 1, O refers to the number of times a given structure was observed during the process, whereas E represents the expected number of occurrences of the structure, which is given in Equation 2.

$$E = \frac{f_1 \cdot f_2}{N} \tag{2}$$

f_1 and f_2 are the number of times the Finnish and English structure has appeared throughout the entire corpus, respectively. For our case, we treat N, the sample size, as the total sum of all observed co-occurrences. By the end of this phase, we would obtain mappings as in the following examples.

We use the results given by the log-likelihood measure to find the most probable English mappings for the 735 most frequent Finnish bigrams in the Bible data. We pick the English n-gram that has the highest score assigned by the method described earlier with an additional limitation that there must be an equal number of placeholders for the same parts-of-speech in the Finnish and the English morphosyntactic structures. The number of punctuations must also be the same and as for closed-class words, the English structure must have at least as many of them as the Finnish structure.

3.2 Filling the Structures for Full Parallel Data

We take a relatively small set of words from the Finnish Wiktionary that are translated into English. This consists of 100 randomly picked words for each open-class part-of-speech (nouns, adjectives, verbs and adverbs) and all of the words registered for the closed class parts-of-speech (adpositions, pronouns, conjunctions, numerals etc). We limit this to a small set of words firstly, to make testing the viability of the approach faster when filling the templates, and secondly to simulate the scarceness of the data of a low-resource language.

We go through all of the parallel morphosyntactic structures and fill in words to match the part-of-speech and morphology in the placeholder. For each parallel structure, we fill it with all the possibilities of different combinations of words to replace the placeholders. We use Syntax Maker [11] together with Omorfi [22] to produce the matching morphological form for the Finnish placeholders. For English, we inflect the words with the pattern library [24].

Additionally, we prepare a parallel dataset which consists purely on the Wiktionary entires. This will make it so that the model sees mappings between closed-class words that didn't necessarily appear in the templates. This data is also input to the model as a part of the training data. Furthermore, we train a baseline model with this dictionary data to see the effect of having the templates on the output. This choice of baseline can be justified by the fact that it is the only kind of parallel data available for a great many low-resource Uralic languages.

4 EVALUATION AND RESULTS

By following the method described in Section 3, we obtain 735 automatically mapped parallel structures. After filling each structure with words from Wiktionary, the method generates 6,673,160 parallel texts.

We evaluate the correctness of the automatically mapped structures by manually annotating them, by an expert linguist who is native in Finnish. We assume that correctly mapped structures would yield correct sentences when filled; hence, we focus the following evaluation on structures.

For the scope of this paper, we randomly select 50 morphosyntactic structures, out of a total of 735 structures. Table 1 shows five randomly selected examples of Finnish morphosyntactic structures and their automatically mapped English structures. The third and fourth columns in the table provide examples of two computationally generated parallel texts by filling out the structures.

From our manual annotation of the automatic mapping of morphosyntactic structures, we report that 54% of the mapped structures were correct. Looking at the examples in Table 1, we can see that the even the incorrect ones such as the verb mapping were only off by morphology and not by part-of-speech. This, however, is not true in the case of incorrectly mapped words such as *ja* and *about* in the third example, for these words can be wildly incorrect as there is no further check on their compatibility when picking out the most suitable mapping.

The final translation model is trained based on the generated data with Open NMT [15]. We built a small corpus of 20 simple Finnish sentences (subject-verb-object plus an occasional adverb and adjective). The words in the sentences consisted of randomly picked words from the Wiktionary sample used to fill the templates. We also translated these sentences into English to calculate BLEU scores.

Translating the sentences as full resulted in some of the words translated correctly while other words being dropped out entirely for every sentence. This, we believe, is due to the fact that the NMT model has never learned a language model of full sentences for English and is therefore not capable of producing one either. This is why we split the test sentences into bigrams and did the translation on that level.

We calculate the BLEU scores of the trained models using the implementation provided in NLTK [5]. We calculate two types of BLEU scores, BLEU-2 and BLEU-4. In BLEU-2, the score is calculated on cumulative bigrams, i.e. unigrams and bigrams have a 50% weight, whereas BLEU-4 is calculated on cumulative 4-grams, i.e. each n-gram has a weight of 25%. While calculating both types of scores, we apply a smoothing as proposed by Lin and Och [16]. Figure 1 shows the BLEU scores of both models, the baseline and the final translation model.

The BLEU scores depicted in Figure 1 show a clear improvement to the baseline in the translation task of bigrams. However, looking at the result more qualitatively in Table 2, we can see that the biggest shortcoming of the model is in grammaticality of the output. It manages to translate the input by correct words, but their surface form realization doesn't follow the English grammar.

Comparing the number of generated texts using our proposed method, i.e. 6,673,160, to the number of sentences in the baseline data, 1,030, we notice that our method is an effective way of producing a huge number of new parallel texts even with a limited vocabulary. This shows in the BLEU scores and in the coverage of different morphological forms.

5 DISCUSSION AND FUTURE WORK

From our initial results, we saw that the model is capable of learning to translate correctly smaller structures. However, it learns very little from the structure of full sentences in the target language which results in a failure to produce syntactic results. This finding

Finnish	English	Finnish Filled Example	English Filled Example
ADJ_Case=Nom_Degree=Pos_Number=Plur ja	ADJ_JJ and	relatiiviset ja	relative and
		harhaanjohtavat ja	misleading and
, NOUN_Case=Nom_Number=Plur	NOUN_NNS ,	, titaanit	titans ,
		, adjektiivit	adjectives ,
ADJ_Case=Ela_Degree=Pos_Number=Sing ja	about ADJ_PRP	luodinkestävästä ja	about bulletproof
		legendaarisesta ja	about legendary
, jotka	, those	–	–
		–	–
VERB_Mood=Cnd_Number=Sing_Person=3 _VerbForm=Fin_Voice=Act .	VERB_VB ,	härnäisi .	tease ,
		aloittaisi .	start ,

Table 1: Random examples of Finnish morphosyntactic structures and their automatically mapped English structures, along with automatic sentences generated by filling the structures.
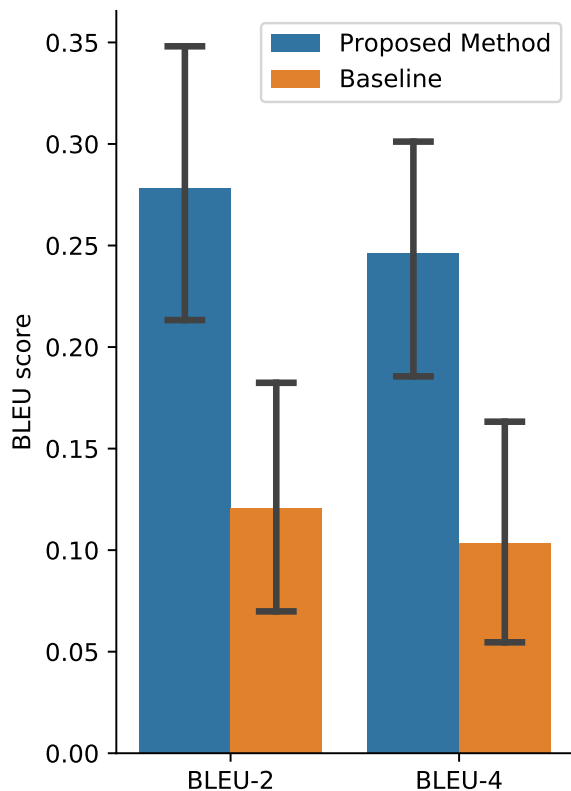

Figure 1: BLEU-2 and BLEU-4 scores of the final translation model and the baseline model.

Input	Predicted Translation	Correct Translation
viikunakottarainen härnää	Purple Glossy Starling tease	a Purple Glossy Starling teases
merivirta ristiinnaulitsee	ocean current crucify	an ocean current crucifies
he vakiinnuttavat	I establish	they establish
ääri lenkkeilee	jogs verge	a verge jogs

Table 2: Examples of the translations

suggests that while the template approach seems to give promising results, the templates should be of longer chains of n-grams, possibly even full sentences in order for the model to learn to produce a desirable output. Therefore, we suggest looking into a more generative framework such as CFG (context-free grammar) in building the templates.

Just having the bigrams was enough to provide some alleviation to the issue rising from the rich morphology of Finnish. The NMT model was capable of translating non-lemmatized word forms into English, although losing their grammatical significance in the process.

This work can be further expanded in the context of Finnish with the abundance of parallel Finnish-English data available by generating full parallel sentences instead of n-grams. However, this might be more challenging in the context of a low-resource language where no automated methods can be applied to extract parallel syntactic structures. However, a way of overcoming this might be incorporating a parallel generative grammar approach to produce a multitude of different sentences with a limited amount of rules. This could, for example, be achieved by tailoring a parallel context free grammar.

In order to make the parallel data more semantically coherent, and to cater for the problems rising from polysemy in a multilingual setting, semantic information could be used. This notion is something that is being used and heavily exploited in the field of computational creativity (c.f. [1, 8]). Furthermore, promising results have been obtained in research using a genetic algorithm for generating parallel data for an NMT model [2, 10].

Additionally, the method for automatically mapping Finnish morphosyntactic structures into the corresponding English structures would benefit from exploring other strongly associated structures. In this paper, we have limited the mapping to only one structural

correspondence, although, in the reality languages exhibit a greater number of parallel syntactic structure that are equivalent in their expression of meaning.

The work presented in this paper is a first step towards the direction of making machine learning approaches possible for the MT of low-resource languages that, in the Uralic context, rely heavily on RBMT. There is, however, more work left to be done in producing more complex parallel data automatically.

6 CONCLUSIONS

Morphologically rich languages pose tremendous challenges for NMT models because of the vast number of inflectional individual forms for each word not covered in a corpus, no matter how big or small. These challenges are highlighted even more in the case of low-resource languages.

Generating parallel data is an effective way of boosting the results of an NMT model. However, the parallel data needs to be representative of the syntax of the language to capture grammaticality and make it possible for the NMT model to learn a language model for the target language.

As we learned from the study presented in this paper, bigram to n-gram structures are helpful in capturing the morphological richness of a Uralic language, but they are not sufficient for syntactic correctness of the output. Therefore, longer template structures are needed.

Generating parallel data for a low-resource language requires a dictionary, a morphological generator and a vast set of parallel morphosyntactic rules complex enough to generate full sentences. Meeting these criteria gives a start for training an NMT model without any real parallel data.

7 ACKNOWLEDGEMENTS

REFERENCES

[1] Khalid Alnajjar, Hadaytullah Hadaytullah, and Hannu Toivonen. 2018. "Talent, Skill and Support." A Method for Automatic Creation of Slogans. In *Proceedings of the Ninth International Conference on Computational Creativity (ICCC 2018)*. Association for Computational Creativity (ACC), Salamanca, Spain, 88–95. http://computationalcreativity.net/iccc2018/sites/default/files/papers/ICCC_2018_paper_24.pdf

[2] Khalid Alnajjar and Mika Hämäläinen. 2018. A Master-Apprentice Approach to Automatic Creation of Culturally Satirical Movie Titles. In *Proceedings of the 11th International Conference on Natural Language Generation*. 274–283.

[3] Vamshi Ambati. 2012. *Active Learning and Crowdsourcing for Machine Translation in Low Resource Scenarios*. Ph.D. Dissertation. Pittsburgh, PA, USA. Advisor(s) Carbonell, Jaime. AAI3528171.

[4] Vamshi Ambati and Stephan Vogel. 2010. Can Crowds Build Parallel Corpora for Machine Translation Systems?. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (CSLDAMT '10)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 62–65. http://dl.acm.org/citation.cfm?id=1866696.1866706

[5] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

[6] Victor Chahuneau, Eva Schlinger, Noah A. Smith, and Chris Dyer. 2013. Translating into Morphologically Rich Languages with Synthetic Phrases. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1677–1687. http://www.aclweb.org/anthology/D13-1174

[7] Stefan Evert. 2008. Corpora and Collocations. In *Corpus Linguistics. An International Handbook*, A. Lüdeling and M. Kytö (Eds.). Vol. 2. Mouton de Gruyter, Berlin, 1212–1248.

[8] Mika Hämäläinen. 2018. Extracting a Semantic Database with Syntactic Relations for Finnish to Boost Resources for Endangered Uralic Languages. In *Proceedings of the Logic and Engineering of Natural Language Semantics 15 (LENLS15)*.

[9] Mika Hämäläinen. 2019. UralicNLP: An NLP Library for Uralic Languages. *Journal of Open Source Software* 4, 37 (2019), 1345. 10.21105/joss.01345.

[10] Mika Hämäläinen and Khalid Alnajjar. 2019. Modelling the Socialization of Creative Agents in a Master-Apprentice Setting: The Case of Movie Title Puns. In *Proceedings of the Tenth International Conference on Computational Creativity*. 266–273.

[11] Mika Hämäläinen and Jack Rueter. 2018. Development of an Open Source Natural Language Generation Tool for Finnish. In *Proceedings of the Fourth International Workshop on Computational Linguistics for Uralic Languages*. 51–58.

[12] Katri Haverinen, Jenna Nyblom, Timo Viljanen, Veronika Laippala, Samuel Kohonen, Anna Missilä, Stina Ojala, Tapio Salakoski, and Filip Ginter. 2014. Building the essential resources for Finnish: the Turku Dependency Treebank. *Language Resources and Evaluation* 48, 3 (01 Sep 2014), 493–531. https://doi.org/10.1007/s10579-013-9244-1

[13] Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks and Incremental Parsing. *To appear* (2017).

[14] Alina Karakanta, Jon Dehdari, and Josef van Genabith. 2018. Neural machine translation for low-resource languages without parallel corpora. *Machine Translation* 32, 1 (01 Jun 2018), 167–189. https://doi.org/10.1007/s10590-017-9203-5

[15] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics-System Demonstrations*. Association for Computational Linguistics, Vancouver, Canada, 67–72. https://doi.org/10.18653/v1/P17-4012

[16] Chin-Yew Lin and Franz Josef Och. 2004. Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip-bigram Statistics. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics (ACL '04)*. Association for Computational Linguistics, Stroudsburg, PA, USA, Article 605. https://doi.org/10.3115/1218955.1219032

[17] Thomas Mayer and Michael Cysouw. 2014. Creating a massively parallel bible corpus. *Oceania* 135, 273 (2014), 40.

[18] Sjur Moshagen, Jack Rueter, Tommi Pirinen, Trond Trosterud, and Francis M. Tyers. 2014. Open-Source Infrastructures for Collaborative Work on Under-Resourced Languages. , 71–77 pages. http://www.lrec-conf.org/proceedings/lrec2014/workshops/LREC2014-Workshop-CCURL2014-Proceedings.pdf The LREC 2014 Workshop "CCURL 2014 - Collaboration and Computing for Under-Resourced Languages in the Linked Open Data Era".

[19] Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving Machine Translation Performance by Exploiting Non-Parallel Corpora. *Comput. Linguist.* 31, 4 (Dec. 2005), 477–504. https://doi.org/10.1162/089120105775299168

[20] Toan Q. Nguyen and David Chiang. 2017. Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation. *CoRR* abs/1708.09803 (2017). arXiv:1708.09803 http://arxiv.org/abs/1708.09803

[21] Peyman Passban, Qun Liu, and Andy Way. 2017. Translating Low-Resource Languages by Vocabulary Adaptation from Close Counterparts. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 16, 4, Article 29 (Sept. 2017), 14 pages. https://doi.org/10.1145/3099556

[22] Tommi A Pirinen, Inari Listenmaa, Ryan Johnson, Francis M. Tyers, and Juha Kuokkala. 2017. Open morphology of Finnish. http://hdl.handle.net/11372/LRT-1992 LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University.

[23] Jack Rueter and Francis Tyers. 2018. Towards an Open-Source Universal-Dependency Treebank for Erzya. In *Proceedings of the Fourth International Workshop on Computatinal Linguistics of Uralic Languages*. 106–118.

[24] Tom De Smedt and Walter Daelemans. 2012. Pattern for python. *Journal of Machine Learning Research* 13, Jun (2012), 2063–2067.

[25] Omar F. Zaidan and Chris Callison-Burch. 2011. Crowdsourcing Translation: Professional Quality from Non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT '11)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1220–1229. http://dl.acm.org/citation.cfm?id=2002472.2002626

[26] Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer Learning for Low-Resource Neural Machine Translation. *CoRR* abs/1604.02201 (2016). arXiv:1604.02201 http://arxiv.org/abs/1604.02201