

Neural and rule-based Finnish NLP models—expectations, experiments and experiences

Tommi A Pirinen
Universität Hamburg
Hamburger Zentrum für Sprachkorpora
Max-Brauer-Allee 60, D-22765 Hamburg
tommi.antero.pirinen@uni-hamburg.de

Abstract

In this article I take a critical look at some recent results in the field of neural language modeling of Finnish in terms of popular shared tasks. One novel point of view I present is comparing the neural methods' results to traditional rule-based systems for the given tasks, since most of the shared tasks have concentrated on the supervised learning concept. The shared task results I re-evaluate, are morphological regeneration by SIGMORPHON 2016, universal dependency parsing by CONLL-2018 and a machine translation application that imitates WMT 2018 for German instead of English. The Uralic language used throughout is Finnish. I use out of the box, best performing neural systems and rule-based systems and evaluate their results.

Tiivistelmä

Tässä artikkelissa tarkastelemme joitain hiljattaisia tuloksia niinkutsutuissa shared task -kilpailuissa suomen kielen hermoverkkomallien osalta. Yksi tämän artikkelin kontribuutioista on hermoverkkomallien tuottamien tulosten vertailu perinteisiin sääntöpohjaisiin kielimallituloksiin, sillä shared task -kisailut pääosin keskittyvät täysin tai osittain hallitsemattomien mallien oppimisen konseptiin. Shared taskit joita tässä artikkelissa tarkastelemme ovat SIGMORPHONin 2016 morfologisen uudelleengeneroinnin kisa, CONLL:n 2018 jäsennyskilpailu sekä WMT 2018:n konekäännöskilpailu uudelleensovellettuna saksan kielelle. Uralilainen kieli jota käytämme kaikissa kokeissa on suomi. Järjestelmät joita käytetään ovat avoimen lähdekoodin järjestelmiä jotka ovat olleet parhaita näissä kilpailuissa.

1 Introduction

The popularity of the neural networks in natural language processing is at the moment climbing very rapidly to the extent that we commonly get to hear that non-neural

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: <http://creativecommons.org/licenses/by/4.0/>

methods should be abandoned. While naturally the majority of this hype is based on English-centric or mostly European NLP, there are some reports of good successes within the less resourced and more morphological languages, including Uralic languages. In this paper I compare directly the state-of-the-art methods between the neural and rule-based language processing for Finnish. I specifically devised experiments based on the following shared tasks and popular systems:

- Generating morphology: Sigmorphon 2016 results (Cotterell et al., 2016) vs. omorfi (Pirinen, 2015a)
- Parsing of morphosyntax: Turku neural parser (Kanerva et al., 2018) vs. omorfi (Pirinen, 2015a)
- Machine translation between Finnish and German: OpenNMT (Klein et al., 2017) vs. apertium-fin-deu (Pirinen, 2018)

Comparing a few different tasks gives us a good overview of the state of the art in the neural processing of Finnish. Parsing tasks give an idea of the potential usability of the language models in various linguistic tasks, such as corpus annotation, whereas the machine translation task provides an important view on the full capacity of the models for a more wide-ranging language understanding task.

One of the contributions of this paper is to gain more insight of the similarities and differences of the traditional rule-based systems for the given tasks, since the shared tasks are virtually always earmarked for more or less supervised language learning, any evaluations between the neural and the rule-based systems are not so commonly found in the literature.

The rest of article is organised as follows: in Section 2, I introduce the shared tasks and rule-based systems at large, in Section 3, I describe the systems used for the experiments, in Section 4, I describe the system setup, in Section 5, I go through the experiments and results, in Section 6, I perform the error analysis, in Section 7, I relate the results to current state of the art as well as practical usage and development of the systems and finally in Section 8, I summarise the findings.

2 Background

In recent years the neural network-based systems, especially so-called deep neural systems, have been brought forward as a solution to all natural language processing problems. Some results have also been provided for Uralic languages. In the case of morphology, there was a popular task of morphological generation as a shared task of the ACL 2016 SIGMORPHON workshop (Cotterell et al., 2016), which included the Finnish generation, and showed some very promising results. In the context of the machine translation, the shared task of the WMT conference has had a Finnish task since 2015, and since 2017 the participants have predominantly been the neural systems (e.g. for 2018 c.f. Bojar et al. (2018)). For the morphosyntax, the popular shared task to test a parser with, is the CONLL task on the dependency parsing (Zeman et al., 2018). What is common with these shared tasks, is that they are aimed for supervised learning of such language models, while in the Uralic NLP the predominant methodology is rule-based, expert-written systems (Moshagen et al., 2014). In this article, I take a practical comparison of building and using the systems for the given tasks as well as a tool in actual linguistic research.

Dictionary	Words	Rules
Omorfi	445,453	58
Apertium-fin-deu	13,119	93

Table 1: Size of the dictionaries in rule-based systems.

3 Methods and Datasets

Omorfi¹ is a lexical database of Finnish, that can be compiled into a finite-state automaton for efficient parsing. Omorfi has wide support for morphological analysis and generation (matching the SIGMORPHON task of morphological regeneration) and parsing (matching the CONLL task for parsing). Apertium-fin-deu² is a hand-crafted rule-based machine translation system based on omorfi, with an addition of a bilingual dictionary and some sets of bilingual rules. This can be used with the apertium tools to translate between German and Finnish.

The default mode of operation in a rule-based system is often based on the concept of *all possible hypotheses*, this is in contrast to shared tasks, which are based on 1-best parsing instead; measuring the results is based on only a single hypothesis per token. To bridge this gap between rule-based morphology and shared tasks, I have used a combination of popular strategies implemented with python scripting language.³ These strategies build in principle on both constraint grammar (Karlsson, 1990; Pirinen, 2015b) and my previous experiences with unigram models in rule-based morphologies (Lindén and Pirinen, 2009), it may, however, be noteworthy that at the time of the writing the solution described is very much a work in progress, so it should not be understood as having any specific advances over the above-referred previous experiments yet. Furthermore, to perform the SIGMORPHON and CONLL tasks I have written small python scripts to analyse and map the analyses between omorfi’s formats and theirs. For machine translation I use the `apertium` command and discard the debugging symbols. Examples of the output mangling we perform can be seen in listing 1. As can be seen in the example, the token 7 (*2017*) has no rule-based dependency analysis, since it is not covered by the very basic dependency labeling script we use.

Some statistics of the rule-based dictionaries can be seen in the table 1.

The *Turku neural parsing pipeline* (referred from now on to as TNPP)⁴ is a recent, popular parser for a language-independent parsing of the dependency structures. They ranked highly in the 2018 CONLL shared task. For the experiments of this paper, I have downloaded the system following the instructions and have not changed any hyperparameters. The model used is `fi_tdt`.

OpenNMT⁵ is one of the many popular neural systems for machine translation. For these experiments I chose it because it provides usable python bindings and it seemed most robust in our early experiments.

The training was performed based on the instructions in the OpenNMT README⁶

¹<https://flammie.github.io/omorfi/>

²<https://apertium.github.io/apertium-fin-deu>

³<https://github.com/flammie/omorfi/tree/develop/src/python>

⁴<https://turkunlp.github.io/Turku-neural-parser-pipeline/>

⁵<https://github.com/OpenNMT>

⁶<https://github.com/OpenNMT/OpenNMT-py#quickstart>

```

$ omorfi-analyse-text.sh -X test/test.text
Juankosken WORD_ID=Juankoski UPOS=PROPN PROPER=GEO NUM=SG CASE=GEN
Juankosken WORD_ID=juan UPOS=NOUN SEM=CURRENCY NUM=SG CASE=NOM
BOUNDARY=COMPOUND WORD_ID=koski UPOS=NOUN NUM=SG CASE=GEN

kaupunki WORD_ID=kaupunki UPOS=NOUN NUM=SG CASE=NOM

liittyy WORD_ID=liittyä UPOS=VERB VOICE=ACT MOOD=INDV TENSE=PRESENT PERS=SGO
liittyy WORD_ID=liittyä UPOS=VERB VOICE=ACT MOOD=INDV TENSE=PRESENT PERS=SG3

Kuopion WORD_ID=Kuopio UPOS=PROPN PROPER=GEO NUM=SG CASE=GEN
Kuopion WORD_ID=kuopia UPOS=VERB VOICE=ACT MOOD=OPT PERS=SG1 STYLE=ARCHAIC

kaupunkiin WORD_ID=kaupunki UPOS=NOUN NUM=SG CASE=ILL

vuoden WORD_ID=vuoden UPOS=ADV
vuoden WORD_ID=vuosi UPOS=NOUN NUM=SG CASE=GEN

2017 WORD_ID=2017 UPOS=NUM SUBCAT=DIGIT NUMTYPE=CARD
2017 WORD_ID=2017 UPOS=NUM SUBCAT=DIGIT NUMTYPE=CARD NUM=SG CASE=NOM

alussa WORD_ID=alku UPOS=NOUN NUM=SG CASE=INE
alussa WORD_ID=alunen UPOS=NOUN NUM=SG CASE=ESS STYLE=ARCHAIC
alussa WORD_ID=alussa UPOS=ADP ADPTYPE=POST
alussa WORD_ID=alussa_2 UPOS=ADV

. WORD_ID=. UPOS=PUNCT BOUNDARY=SENTENCE

$ omorfi-tokenise.py -a src/generated/omorfi.describe.hfst -O conllu -i
test/test.text |
omorfi-conllu.py -a src/generated/omorfi.describe.hfst
--not-rules src/disamparsulation/omorfi.xml

# new doc id= test/test.text
# sent_id = 1
# text = Juankosken kaupunki liittyy Kuopion kaupunkiin vuoden 2017 alussa.
1 Juankosken Juankoski PROPN N Case=Gen|Number=Sing 2 nmod:poss
_ Weight=0.01
2 kaupunki kaupunki NOUN N Case=Nom|Number=Sing 3 nsubj _
Weight=0.005
3 liittyy liittyä VERB V
Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|Voice=Act 0
root _ Weight=0.21000000000000002
4 Kuopion Kuopio PROPN N Case=Gen|Number=Sing 5 nmod:poss _
Weight=0.01
5 kaupunkiin kaupunki NOUN N Case=Ill|Number=Sing 3 obl _
Weight=0.01
6 vuoden vuosi NOUN N Case=Gen|Number=Sing 3 obj _ Weight=0.015
7 2017 2017 NUM Num NumType=Card _ _ _ Weight=500.0
8 alussa alku NOUN N Case=Ine|Number=Sing 3 obl _ Weight=0.025
9 . . PUNCT Punct _ 3 punct _ Weight=0.03

Same output is directly generated by TNPP:

$ cat ~/github/flamie/omorfi/test/test.text |
python3 full_pipeline_stream.py --conf models_fi_tdt/pipelines.yaml --pipeline parse_plaintext
# newdoc
# newpar
# sent_id = 1
# text = Juankosken kaupunki liittyy Kuopion kaupunkiin vuoden 2017 alussa.
1 Juankosken Juankoski PROPN N Case=Gen|Number=Sing 2 nmod:poss _
_
2 kaupunki kaupunki NOUN N Case=Nom|Number=Sing 3 nsubj _ _
3 liittyy liittyä VERB V
Mood=Ind|Number=Sing|Person=3|Tense=Pres|VerbForm=Fin|Voice=Act 0
root _ _
4 Kuopion Kuopio PROPN N Case=Gen|Number=Sing 5 nmod:poss _ _
5 kaupunkiin kaupunki NOUN N Case=Ill|Number=Sing 3 obl _ _
6 vuoden vuosi NOUN N Case=Gen|Number=Sing 8 nmod:poss _ _
7 2017 2017 NUM Num NumType=Card 6 nummod _ _
8 alussa alku NOUN N Case=Ine|Derivation=0|Number=Sing 3 obl _
SpaceAfter=No
9 . . PUNCT Punct _ 3 punct _ SpacesAfter=\n

```

Figure 1: Example of omorfi’s outputs and the shared-task equivalents converted.

Corpus	Sentences
Europarl train	1,768,817
dev	1620
test	1620

Table 2: Size of the corpora in sentences

and no additional hyperparameter-tuning was performed. The training was based on europarl version 7 (Koehn, 2005), pre-processed as suggested on their website⁷. The resulting corpus is summarised in Table 2.

4 Experimental setup

An interesting part of this experiment is the setup, since one of the aspects we present in this paper is usability testing of the neural vs. traditional methods for use of an average Computational Uralist, I also want to get a feel of the *user experience* (UX).

The system setup for all the systems is quite similar, all the free and open software used in these experiments are hosted by github. After cloning, the traditional rule-based systems rely on classical command-line installations, this means that user is expected to install dependencies the best they see and then run compilation of the data using `configure` and `make` scripts, and neural systems use `python` equivalents. In terms of dependencies, all systems are basically well covered with some easy way to install necessary dependencies with single command, such as `pip` or `apt-get`. A bit like rule-based systems, the neural systems need to “compile” i.e. learn neural network binaries from large data, in practice the experience for the end user is the same, except for the wait time, which is slightly longer for the neural-based systems. For Finnish analysers an option is provided to download readily compiled models, while for translation models there is no option. This is equally true for both neural and rule-based models. To parse or translate I have run the systems with default / suggested settings.

To get an idea of intended mode of use (instant, batch processing over the weekend) of the systems and steps, I have collected some of our usage times in the table 3. The real bottleneck for our experiments was the neural machine translation training time, the multi-day training period is problematic in itself, but it is also fragile enough that minor impurities in parallel corpus may ruin the whole model which means that on typical use case user may need to train the model multiple times before reaching to a functional one.

To know how much time to create a system takes from scratch it is also useful to know the amount of data is needed to build it; for rule-based systems this is the size of dictionary, and rule-sets, for neural system it is the training data set size. Both of these factors are especially interesting for Uralistic usage, since the availability of free and open data is rather scarce. The dictionaries are summarised in Table 1 and the corpora in Table 2

For my OpenNMT setup I have created an autotools-based model builder / test runner, that is available in github for repeatability purposes⁸.

⁷<https://statmt.org/europarl/>

⁸<https://github.com/flammie/autostuff-moses-smt/>

Phase, System	omorfi	TNPP	apertium	OpenNMT
Compiling	15 minutes	—	40 seconds	>5 days
(Downloading)	yes	yes	no	no
Parsing / translating	5 minutes	10 minutes	5 seconds	30 minutes
(Speed)	5 sents/s	3 sents/s	324 sents/s	0.9 sents/s
Model size	25 MiB	770 MiB	33 MiB	7420 MiB

Table 3: Usage times of rule-based and neural systems, time-units are indicated in the table. For TNPP I have found no documentation on how to repeat model building or what time it has taken. Sents/s stands for average sentences per second. Model sizes gives you the total size of binaries on disk in binary-prefixed bytes (by 1s -h).

Test set	Baseline	Winning system	Omorfi
Task 1	64.45	97.30	93.92
Task 2	59.59	97.40	93.20
Task 3	56.95	96.56	92.18

Table 4: 1-Best precisions for SIGMORPHON shared task 2016 in Finnish, the winning Neural system and omorfi scores.

5 Evaluation

I present an evaluation of the systems using the standard metrics from the shared tasks.

For morphological generation, the shared task was evaluated by measuring average precisions over all languages, for this experiment I compare the results for Finnish on 1-best predictions only, as I am interested in specific comparison relevant for a single Uralic language. The results are summarized in table 4.

For morphosyntactic analysis the standard evaluations would be based on attachment scores, however, the rule-based system only creates partial dependency graphs with potentially ambiguous roots; this does not work with the official evaluation scripts, so I provide instead a raw 1-best precision result for the specific fields in the CONLL-U format. The results are shown in table 5; The *lemma* row corresponds 3rd CONLL-U column, *UPOS* 4th, *Ufeats* 6th, *XPOS* 5th, *Dephead* 7th, and *Deplabel* 8th. The match is made on strict equality on the string comparison of the whole content, i.e. no re-arranging or approximate matching is performed.

For machine translation the standard shared task evaluation method is to use well-known metrics that compare translations to reference, specifically BLEU. In table 6 I measure the BLEU scores for europarl translations.

6 Error Analysis

As a general trend I see that the precision of the neural systems as well as the BLEU score of the neural machine translation are above of the rule-based systems. I also wanted to know if there is any systematicity to the errors, that the different approaches make. Interesting way forward would be to gain some insight on how the errors for each system could be fixed if at all. One of the commonly mentioned advan-

Column	Turku Neural parsing pipeline	Omorfi
Lemma	95.54	82.63
UPOS	96.91	83.88
Ufeats	94.61	73.95
XPOS	97.89	89.58
Dephead	90.89	33.13
Deplabel	92.61	49.01

Table 5: 1-best precisions of Turku neural parsing system and omorfi. The numbers were measured with our script since the official test script does not handle partial dependency graphs or multiple roots.

Language pair	OpenNMT	Apertium
German to Finnish	7.09	0.6
Finnish to German	7.12	0.3

Table 6: Automatic translation evaluations, metrics from WMT shared tasks 2018 and corpora from europarl evaluation section. BLEU scores have been measured with the tool `mteval-14.perl`.

tages of a rule-based system is that it is predictable and easy to fix or extend; whether a missing form in generation or analysis is caused by a missing word in lexicon, a missing word-form in paradigm or ordering of the alternative forms, the solution is easy to see. With a neural system the possibilities are limited to adding more data or modifying hyperparameters.

When looking at the errors in the morphological regeneration test for rule-based system, I can see several categories emerge: *True OOV* for lexemes missing from the database (e.g. *ovovivivipaarisuus*), *Wrong paradigm* for wordforms that are generated but with some errors, such as wrong vowel harmony or consonant gradation (e.g. *manuskripteiltä* pro *manuskripteilta* (from manuscripts)) and *Real allomorph / homograph* for cases where the correct form is recalled but not at best-1 due to ambiguous lexeme or free allomorphy (for example, I generate *köykistämäisillänsä* pro *köykistämäisillään* (about to defeat), but both are equally acceptable). In the leftover category I found among others, actual bugs in the generation functionality. For example, I was unable to generate the forms of *aliluutnantti* (sub-lieutenant) since the generation function failed to take into account extra semantic tags it contains.⁹ I sampled a total of 65 errors and the results can be seen in the table 7.

In the dependency parsing task one of the most common errors in the rule-based system seems to be the `Person=0` feature with 766 occurrences in the test set, as it is systematically ambiguous with `Person=3` for all singulars, it is probably a true ambiguity in that there are not many context clues to disambiguate it. Another systematic source of errors seems to be the systematic ambiguity between auxiliary and common verbs, which also shows up in the parsing of copula structures and in the morphological features. Similarly, a common problem of rule-based systems in parsing tasks is the etymological systematic ambiguity created by derivation and lexicalisation, that

⁹a bug has been since fixed but I include the original error analysis in the article for an interesting reference

Type	Count	Percentage
OOVs	23	36 %
Wrong paradigm	20	31 %
Allomorphs	9	15 %
Others	13	20 %
Total	65	100%

Table 7: Rule-based morphology generation errors classified. .

affects participle above anything, but also less productive features. It would appear that OOV’s do not contribute here greatly to the error mass, despite consisting total of 460 appearances the baseline guess of singular nominative nominal for the OOV’s is surprisingly often sufficient.

Looking at both rule-based and neural systems for MT, it is easy to tell that for example the OOV’s constitute a large part of errors, and exist in most sentences. Judging the actual translation quality by sampling the sentences also reveals a quality that is overall not sufficient for computer aided translation or gisting, to the extent that I believe further analysis may not be fruitful without further development of the underlying models first.

7 Discussion

One of the goals in this experiment was to find out how usable the neural and traditional models are for a computational linguist who might want to pick a state-of-the-art parser off-the-shelf and use it for text analysis or translation related tasks. Based on my initial impression, I would probably recommend making use of the neural parsers for languages where enough training data is available, and aiming to make training data where it is not. However, for a low resource language, it might often be easier to create a sufficiently large dictionary with rule-based model than to curate realistic corpus and annotate it, and given that the results of a rule-based system are not such far from the state-of-the-art in neural systems for the given metrics, they should be well sufficient for parsing. On top of that, the resources created with a rule-based system are a part of necessary NLP system for language survival (writer’s tools, electronic dictionaries) that neural systems do not offer it does not make sense to put all eggs in one neural network.

One thing that has been left out of the experiment is what is required for developing a new system: dictionaries and grammars for rule-based systems, treebanks or parallel texts for the neural systems. These are available at the moment for the main Uralic languages: Finnish, Hungarian and Estonian, and to smaller extent also for Northern Sámi, Erzya. The question then remains, is it easier for a minority Uralic language to develop a treebank and a parallel corpus, or dictionaries and grammar, or both.

One noteworthy point to the method of developing resources, as well as to our evaluation, is, that the original Turku dependency treebank was in fact developed based on the analyses provided by an old version of omorfi Haverinen et al. (2014),¹⁰ and that was used as a basis for building the UD-Finnish-TDT treebank, that is used

¹⁰we thank the anonymous reviewer for bringing point up

as a model for the TNPP analyser. So a traditional way to build resources for neural parsers still requires an existing high-quality rule-based parser as well as a lot of native human annotation work on the one hand, on the other hand, the combination of rule-based parses and human annotation does result in a parser that is more precise at predicting in basic setup.

One thing that might be a common expectation is, is that a rule-based systems that have been developed for a long time, should score very highly in basic tasks like morphological generation and parsing, since apart from real OOV's and bugs, correctly made morphology should virtually be able to generate 100 % of the word-forms in its dictionary. For the precision of 1-best analyses however, there can be small portion of word-forms that either exhibit unexpected (in terms that writer of rule-based parser had expected form to be ungrammatical) or free variation. For recall, which is typically the first goal for rule-based analyser, the value is nearer to the virtual 100 % (Pirinen, 2015a).

One surprising thing I found out, that when testing the machine translations on a non-English pair, the out-of-the-box results for both approaches are very modest, suggesting that more work is needed to for a usable MT as a tool for Uralist than just picking off-the-shelf product at the moment. While our test was still based on non-Uralic language partly due to resource and time constraints, I believe the results will still give a good indication of the current state-of-the-art. Notably, it is not unlikely for a research group in Uralistics to need machine translations of German or for example Russian as well.

So far, I have only used the precision and BLEU measures to evaluate the systems, it is likely that different metrics would show more favourable results for a rule-based systems that typically maximise recall or coverage first.

One of the surprising finds that I had when fitting the rule-based systems to non-rule-based shared tasks is, is that I could repurpose the task as a new automatic continuous integration test set for the lexical database, and the tests have already proved useful for recognition several types of easily fixable errors in the database. I note that, in the rule-based system fixing the OOV-type errors and the paradigm type errors is typically a trivial fix of one line of code taking less than a minute, however, improving the allomorph selection or homograph disambiguation is an open research question.

For future work I will study both the neural and rule-based systems further with hopefully intra-Uralic pairing as well, to find if it's plausible for actual use.

8 Conclusion

I performed some experiments to find out what is the current state-of-the-art status between neural and rule-based methods for Finnish, I have found out that the neural methods perform admirably for all parsing approaches for the given test sets that they were designed for, but rule-based methods are also still within acceptable distance. For non-parsing task such as machine translation in Uralic languages the methods are probably not yet sufficient to be efficiently used as a tool for research, but further research and development is needed.

Acknowledgments

The author was employed by CLARIN-D during the writing of the article.

References

- Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Philipp Koehn, and Christof Monz. 2018. Findings of the 2018 conference on machine translation (wmt18). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*. Association for Computational Linguistics, pages 272–303. <http://aclweb.org/anthology/W18-6401>.
- Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The sigmorphon 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Association for Computational Linguistics, pages 10–22. <https://doi.org/10.18653/v1/W16-2002>.
- 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):493–531.
- Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. Turku neural parser pipeline: An end-to-end system for the conll 2018 shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics.
- Fred Karlsson. 1990. Constraint grammar as a framework for parsing unrestricted text. In H. Karlgren, editor, *Proceedings of the 13th International Conference of Computational Linguistics*. Helsinki, volume 3, pages 168–173.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*. <https://doi.org/10.18653/v1/P17-4012>.
- Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*. volume 5, pages 79–86.
- Krister Lindén and Tommi Pirinen. 2009. Weighting finite-state morphological analyzers using HFST tools. In Bruce Watson, Derrick Courie, Loek Cleophas, and Pierre Rautenbach, editors, *FSMNL 2009*. <http://www.ling.helsinki.fi/klinden/pubs/fsmnlp2009weighting.pdf>.
- Sjur Moshagen, Jack Rueter, Tommi Pirinen, Trond Trosterud, and Francis M Tyers. 2014. Open-source infrastructures for collaborative work on under-resourced languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC*. pages 71–77.
- Tommi A Pirinen. 2015a. Development and use of computational morphology of finnish in the open source and open science era: Notes on experiences with omorfi development. *SKY Journal of Linguistics* 28.
- Tommi A Pirinen. 2015b. Using weighted finite state morphology with visl cg-3-some experiments with free open source finnish resources. In *Proceedings of the Workshop on “Constraint Grammar-methods, tools and applications” at NODALIDA 2015, May 11-13, 2015, Institute of the Lithuanian Language, Vilnius, Lithuania*. Linköping University Electronic Press, 113, pages 29–33.

Tommi A Pirinen. 2018. Rule-based machine-translation between finnish and german.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. Conll 2018 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*. Association for Computational Linguistics, pages 1–21. <http://aclweb.org/anthology/K18-2001>.