Apertium-fin-eng—Rule-based shallow machine translation for WMT 2019 shared task

Tommi A Pirinen
Universität Hamburg
Hamburger Zentrum für Sprachkorpora
tommi.antero.pirinen@uni-hamburg.de

Abstract

In this paper I describe a rule-based, bi-directional machine translation system for the Finnish—English language pair. The original system is based on the existing data of FinnWordNet, omorfi and apertium-eng. I have built the disambiguation, lexical selection and translation rules by hand. The dictionaries and rules have been developed based on the shared task data. I describe in this article the use of the shared task data as a kind of a test-driven development workflow in RBMT development and show that it suits perfectly to a modern software engineering continuous integration workflow of RBMT and yields big increases to BLEU scores with minimal effort. The system described in the article is mainly developed during shared tasks.

1 Introduction

This paper describes our submission for Finnish—English language pair to the machine translation shared task of the *Fourth conference on machine translation* (WMT19) at ACL 2019. Traditionally *rule-based machine translation* (RBMT) is not in the focus for WMT shared tasks, however, there are two reasons I experimented with this system this year. One is that we have had an extensively large amount of lesser used resources for this pair: omorfi[1] (Pirinen, 2015) has well over 400,000 lexemes[2], apertium-eng[3] has over 40,000 lexemes and apertium-fin-eng[4] over 160,000 lexeme-to-lexeme translations. One of our key interests in the shared task like this is that it provides an ideal data for test-driven development of lexical resources.

One concept I experimented with the shared task is various degrees of automation—expert supervision for the lexical data enrichment. In this experiment I used automatic methods to refine the lexical selection of the machine translation, and semi-automatised workflows for the generation of the lexical data, as well as some expert-driven development of the more grammatical rules like noun phrase chunking and determiner generation. It might be noteworthy that this machine translator I describe in the article is not actively developed outside the shared tasks, so the article is moreso motivated as an exploration of the workflow and methods on semi-automatically generated shallow RBMT than a description of a fully developed RBMT.

The rest of the article is organised as follows: In Section 2 I describe the components of our RBMT pipeline, in Section 3 I describe the development workflow and in Section 4 I show the shared task results and I perform error analysis and discuss the results and finally in Section 5 we summarise the findings.

2 System description and setup

The morphological analyser for Finnish is based on omorfi (Pirinen, 2015), a large morphological lexical database for Finnish. Data from omorfi has been converted into Apertium format and is freely available in the apertium-style format in the github repository apertium-fin[5]. For English I have used Apertium's standard English analyser apertium-eng[6]. Both analysers were downloaded from github in the beginning of the shared task and we have updated and further developed them based on the development data during the shared

[1] https://github.com/flammie/omorfi
[2] https://flammie.github.io/omorfi/statistics.html
[3] http://wiki.apertium.org/wiki/English
[4] https://github.com/apertium/apertium-fin-eng
[5] https://github.com/apertium/apertium-fin
[6] https://github.com/apertium/apertium-eng

Proceedings of the Fourth Conference on Machine Translation (WMT), Volume 2: Shared Task Papers (Day 1) pages 335–341

Dictionary	Lexemes	Manual rules
Finnish	426,425	143
English	40,185	187
Finnish-English	164,501	273

Table 1: Sizes of dictionaries. The numbers are numbers of unique word entries or translation entries as defined in the dictionary, e.g., homonymy judgements have been made by the dictionary writers. The rule counts are combined counts of all sorts of linguistic rules: disambiguation, lexical selection, transfer and so forth.

task. I developed the Apertium's Finnish-English[7] dictionary initially based on the FinnWordNet's translated data, which was over 260,000 Wordnet-style lexical items; of these I discarded most which had multiple spaces in them or didn't match any source or target words in Finnish and English dictionaries, ending with around 150,000 lexical translations. The size of dictionaries at the time of writing is summarized in Table 1, however more up-to-date numbers can be found in Apertium's Wiki [8]

The system is based on the Apertium[9] machine translation platform (Forcada et al., 2011), a shallow transfer rule-based machine translation toolkit. For morphological analysis and generation, HFST[10] (Lindn et al., 2011) is used and for morphological disambiguation VISL CG-3 [11] is used. The whole platform as well as all the linguistic data are licensed under the GNU General Public Licence (GPL).

Apertium is a modular NLP system based on UNIX command-line ideology. The source text is processed step-by-step to form a shallow analysis (morphological analysis), then translated (lexical transfer) and re-arranged (structural transfer) to target language analyses and finally generated (morphological generation). Each of the steps can be processed with arbitrary command-line tool that transforms the input in expected formats. All of the steps also involve ambiguity or one-to-many mappings, that requires a decision, and while these decisions can be made using expert written rules, the writing of the rules is also a demanding task, and it is interesting to see how much can be achieved by simply bootstrapping the rulesets using automatic rule acquisition.

To illuminate how apertium does RBMT in Finnish—English, and the kinds of ambiguities I resolve, I show in Table 3 examples of the ambiguities with an example sentence. The ambiguity of source morphology is the true ambiguity rate of the language (according to the morphological analyser), i.e. how many potential interpretation each word has. It is no surprise that Finnish has relatively high ambiguity rate, however, English is nearly unambiguous is more due to limitation of apertium's English dictionary than feature of English per se, given that English has a bit of productive zero-derivations, e.g. verbing nouns and vice versa. The lexical selection ambiguity is the translation dictionary's rate of choices per source word, and FinnWordNet on average has 5 synonyms per word to suggest. The target morphology ambiguity is the rate of allomorphy or free variation, in Finnish as target language there's some systematic problems, such as plural genitives and partitives, whereas English literally has two incidents in the whole dev set: *sown / sowed* and *fish / fishes*. Assuming a perfect RBMT system would keep all options open, until final decision, the number of hypotheses at the end would be at least $\mathrm{MA_{SL}} \times \mathrm{LS_{SL \to TL}} \times \mathrm{MA_{TL}}$, where MA is morphological ambiguity rate, LS is lexical selection ambiguity rate, $_{SL}$ is source language and $_{TL}$ is target language. For Finnish—English I show the example figures of the ambiguities based on the development and test sets in Table 2.

The rule-based machine translation process as it is performed by apertium is shown in Table 3. The first step of the RBMT here is morphological analysis, in apertium this covers both tokenisation and morphological analysis as seen here; in apertium-eng the expression 'in front of' is considered to be a single token and is packaged as a preposition (we have also omitted an ambiguity between attributive and nominal reading of the house, since the distinction does not currently make difference in English to Finnish translation, in order to fit the table in the paper). The morphological analysis in apertium is performed by finite-state morphological analysis as defined in Beesley and Karttunen (2003) and implemented in open source format

[7] https://github.com/apertium/apertium-fin-eng
[8] http://wiki.apertium.org/wiki/List_of_dictionaries
[9] https://github.com/apertium
[10] https://hfst.github.io
[11] http://visl.sdu.dk/cg3.html

Feature: Corpus	**Source morphology**	**Lexical** selection	**Target** morphology	*Total*
Finnish dev set	1.68	5.04	1.0002	8.46
Finnish test set	1.69	4.80	1.0003	8.13
English dev set	1.04	1.15	1.0013	1.19
English test set	1.03	1.12	1.0006	1.15

Table 2: Ambiguity influencing RBMT Finnish-to-English and English-to-Finnish

by Lindn et al. (2011). After analysis, the next step is to disambiguate, i.e. pick 1-best lists of morphological analyses; in apertium this is done by constraint grammar, as described by Karlsson (1990) and implemented in open source by VISL CG 3.[12]. In lexical translation phase, each lemma is looked up from the translation dictionary, and in lexical selection the translation that is most suitable by the context and statistics is selected. In the structural transfer phase a number of things is performed: the English morphological analyses are rewritten into Finnish analyses, e.g. the adjective and noun will receive a genitive case tag due to the adposition, and the adposition is moved before the noun phrase since it is a preposition in Finnish and postposition in English, and the article is just removed, as the use of articles is non-standard in Finnish. Finally the Finnish analysis is generated into a surface string using a finite-state morphological analyser, since they are inherently bidirectional this needs no extra software or algorithms.

3 RBMT development workflow

I present here different levels of automation in the RBMT workflow: in Subsection 3.1 I have automated the generation of rules, in Subsection 3.2 I have a semi-automated workflow and finally in Subsection 3.3 I have an expert-driven development workflow.

3.1 Lexical selection training

One of the key components of this experiment was to try automatic rule-creation mechanisms for the converted Wordnet dictionary refinement. A large number of translation quality issues in the initial converted Wordnet dictionary was a high number of low-frequency 'synonyms' in translations. To overcome this some automatic methods were used. For automatic bootstrapping of the lexical selection rules I used Europarl corpus (Koehn, 2005) data and the methods demonstrated by Tyers et al. (2012). Since the result of this training seemed also insufficient, I experimented with another system to generate more rules for lexical selection.[13] On top of that, I have updated the lexical selection with some manual rules, that were either not covered by Europarl hits or skewed wrongly for the news domain, for example, the word 'letter' seemed to mainly have translations of *kirje* (a message written on paper), while in the development set all the sentences I sampled, a more suitable translation would of been *kirjain* (a character of alphabet). The resulting lexical selection rule sets are summarised in the table 4. The first method of creating rules is based on n-gram patterns, due to restricted time and processing resources I have only included bigrams into this model, and the second model only considers unigrams. The results are added up in the table lines + *bigrams* and + *unigrams* respectively.

3.2 Lexicon development workflow

One of the key components of this experiment is to show that a *shared-task driven development* (STDD) is a usable workflow for the development of the lexical data in rule-based machine translation system. As such, a 'training' phase in the RBMT development has been replaced by a very simple semi-automated native speaker - driven project workflow consisting of following:

1. Collect all lexemes unknown to source language dictionary, and add them with necessary morpholexical information

2. Collect all lexemes unknown to bilingual translation dictionary, and add their translations

[12] http://visl.sdu.dk/cg3.html

[13] https://svn.code.sf.net/p/apertium/svn/trunk/apertium-swe-nor/dev/lex-learn-unigram.sh

Input:	In front of the big house
Morphological analysis:	In front of.PREP the.DET.DEF.SP big.ADJ house.N.SG
Morphological disambiguation:	In front of.PREP the.DET.DEF.SP big.ADJ house.N.SG
Lexical translation:	In front of.PREP→Edessä.POST the.DET.DEF.SP→se.DET.DEF.SP big.ADJ→iso∼raju∼paha∼kova∼...jalomielinen.ADJ house.N.SG→huone∼talo∼suku∼...edustajainhuone.N.SG
Lexical selection:	Edessä.POST se.DET.DEF.SP iso.ADJ talo.N.SG
Structural transfer:	iso.ADJ.POS.SG.GEN talo.N.SG.GEN Edessä.POST
Finnish translation:	ison talon Edessä

Table 3: Translation process for the English phrase 'In front of the big house'

Orig. Fin-Eng	18,066
+ bigrams	24,662
+ unigrams	30,049
Orig. Eng-Fin	22
+ bigrams	24,631
+ unigrams	25,748

Table 4: Lexical selection rules statistically generated

3. Collect all lexemes unknown to the target language dictionary, and add them to the dictionary with necessary morpholexical information

The semi-automation that I have developed lies in collecting the different unknown lexemes or *out-of-vocabulary* items (OOVs), and guessing a lexical entry or multiple plausible entries for them and have the dictionary writer select and correct them.

3.3 Grammar development

An expert-driven part of the RBMT workflow in our current methodology is the grammar development. This consists manually reading the sentences produced by the MT system to spot systematic errors caused by grammatical differences between languages. For the purposes of this shared task and the workshop, the linguistics or grammar are not a central concept, so I will not detail it here in detail. In practice this concerns of such grammatical rules as mapping between no articles in Finnish to articles in English, mapping between case or possessive suffixes and their corresponding lexical representations in English and so forth.

Corpus	BLEU-cased	CharacTER
MSRA.NAO	27.4	0.515
HelsinkiNLP RBMT	8.9	0.650
apertium-eng-fin	4.3	0.756
USYD	33.0	0.494
apertium-fin-eng	7.6	0.736

Table 5: automatic scores from http://matrix.statmt.org, we show our scores (boldfaced), the highest ranking RBMT and the highest ranking NMT for reference.

The details can be seen in the code that is available in github.

4 Evaluation, error analysis and discussion

The automatic measurements as used by the shared task are given in the table 5. I show here the BLEU (Papineni et al., 2002) and the CharacTER scores. BLEU, as it is a kind of industry standard, and CharacTER (Wang et al., 2016) as it is maybe more suited for morphologically complex languages. As the automatic scores show, the rule-based system has still room for improvement.

I find that a linguistic error analysis is one of the most interesting part of this experiment. The reason for this is is that the experiment's scientific contribution lies more in the extension of linguistic resources and workflows than machine learning algorithm design. It is noteworthy, that in the sustainable workflow I demonstrate in this article, error analysis is a part of the workflow, namely, adding of the lexical data and rules follows the

Error	count
OOVs in Finnish	763
OOVs in English	943
OOVs in FinEng	2696

Table 6: Classification of mainly lexical errors in apertium-fin-eng submissions for 2019

Corpus	BLEU-cased
apertium-eng-fin 2015	2.9
2017	3.5
2019	4.3
apertium-fin-eng 2015	6.9
2017	6.3
2019	7.6

Table 7: Progress of apertium-fin-eng over the years using only the WMT shared task driven development method.

layout given in Section 3 and is the same for development and error analysis phase. I have, to that effect, categorised the errors in translations along the workflow:

1. OOV in source language dictionary (including typos and non-words)
2. OOV in bilingual dictionary
3. OOV in target language dictionary
4. disambiguation or lexical selection fail
5. structural failure or higher level

The OOV's can be calculated automatically from the corpus data, but the higher level failures need human annotation. A summary of the errors can be seen in the table 6, this is based on the errors that were fixed as a part of error analysis process. As a result of this workflow, I have improved the BLEU points of apertium-fin-eng over the years, as can be seen in the table 7.

The OOV numbers might look moderately large but a major part falls under proper nouns, which are generally low frequency and do not cause a large problem in translation pipeline, the untranslated proper noun is recognisable and the mapping of adpositions and case inflections will fail where applicable. The task of adding proper nouns to the dictionaries is also simplest, they are easy to gather from the text, and for English and bilingual dictionaries no further classification is necessary; for the Finnish dictionary entry generation, paradigm guessing is necessary, although the paradigms used in foreign names are much more limited than with other parts-of-speech to be added. In the *newstest 2019* data there was a number of words that I decided not to add to our dictionaries, unlike our usual workflow where I aim at virtual 100 % coverage with gold corpora. The unadded words were for example words like "Toimiluvanmuodossatoteutettavajulkisenjayksityisensektorinkumppanuus", which seems to have a large number of missing spaces and extra hyphen, these as well as extraneous spaces were quite common in the data in our error analysis as well as 'words' like 'OIet', 'OIi', 'OIin', 'OIisi', 'OIIut', i.e. forms of 'olla' (to be) where lowercase L has been replaced with uppercase I. While I do account for common spelling mistakes in our dictionaries, these kind of errors are probably more suited for robustness testing and implemented with spelling correction methods for specific problematic generated text, such as OCR. We will look into implementing spelling correction into our pipeline in the future. Comparing the performance of RBMT to NMT, it can be clearly seen that contemporary NMT is better suited for error tolerance, in part because it can be more character-based than token-based, in part because any large training data set will actually have some OCR errors and run-in tokens.

After OOV-errors one of the biggest easily solvable problems is ambiguity, so word sense disambiguation and lexical selection. For lexical selection I found about 200 lexical translations that were still badly wrong and could be solved without coming up complex context conditions. For disambiguation problems, a surprisingly common problem was sentence-initial proper noun that is a common noun as well, as a high frequency example, for the word 'trump' meaning a winning suit in card games (= Finnish 'valtti') would get selected over the POTUS, plausibly when most of the training and development before WMT 2019 did not contain so many proper noun Trumps. Also rather common problem still is the ambiguity in English verb forms, and between English zero derivations.

In the structural transfer a large number of errors are caused by long-distance re-ordering.

For example for Finnish to English proper noun phrases regardless of length of the phrase, the Finnish shows case in last word or postposition after the last word, English has preposition before the word, but when phrase gets chunked partially the adpositions or case suffixes end up in the middle with a rather jarring effect to the translated sentence. The same applies for other effects where generating correct language depends on correct chunk detection, e.g. the article generation is very limited in the current code because the articles need to be generated from nothing, when translating from Finnish to English, only at the very beginning of specific noun phrases.

Finally a number of problems were caused for such grammatical differences between languages that do not have a good solution in lexical rule-based machine translation, such as difference between English noun phrases and corresponding Finnish compound nouns or for example the common English class of -able suffixed adjectives that does not have accurate lexical Finnish translation at all.

In terms of where RBMT is perhaps more usable than NMT, one important factor is how predictable and systematic the errors are when they appear. For example just looking at the first page of the top-ranking system in Finnish-to-English[14] one can see the Finnish "Aika nopeasti saatiin hommat sovittua, Kouki sanoi" translated into "Pretty quickly we got the gays agreed, Kouki said." whereas the correct translation is "We reached a pretty quick agreement, Kouki said.", the big problem with the neural translation is that it is deceptively fluent language but conveys something completely different, comparing to the rule-based version: "Kinda swiftly let jobs agreed, Kouki said." which is not fluent at all, but doesn't hallucinate gays there so it may be more usable for post-editing. For further research in the problems of NMT for real-world use, see for example Moorkens et al. (2018).

In comparison to neural and statistical systems, the rule-based approach does not generally fare well as measured with automatic metrics like BLEU, for a human evaluation refer to (Bojar et al., 2019). However, the experiment I describe here is also not the most actively developed machine translators, rather I use the experiment to gauge the effects the described workflow has to quality of semi-automatically generated RBMT, to see how more developed systems fare on the same task you should also refer to (Hurskainen and Tiedemann, 2017; Kolachina and Ranta, 2015).

[14] http://matrix.statmt.org/matrix/output/1903?score_id=39757

5 Concluding remarks

In this article I've shown a workflow of *shared task driven development* for rule-based machine translations, namely the lexicons and rules. I show that a small effort to update lexical data based on yearly released gold corpora increases BLEU points and enlarges dictionaries as well as improves rulesets sizes and qualities by a significant amount. In future I aim to build more automatisation for the workflow to make it trivially usable with continuous integration.

The systems are all available as free/libre open-source software under the GNU GPL licence, and can be downloaded from the internet.

Acknowledgements

This work has been written while employed in the *Hamburger Zentrum für Sprachkorpora* by CLARIN-D. Thanks to all contributors of the related projects: omorfi, FinnWordNet, Apertium, and everyone helping with apertium-fin, apertium-eng and apertium-fin-eng.

References

Kenneth R Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI publications.

Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp +Koehn, Christof Monz, Mathias Müller, and Matt Post. 2019. Findings of the 2019 conference on machine translation (wmt19). In *Proceedings of the Fourth Conference on Machine Translation, Volume 2: Shared Task Papers*, Florence, Italy. Association for Computational Linguistics.

Mikel L Forcada, Mireia Ginestí-Rosell, Jacob Nordfalk, Jim ORegan, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez, Gema Ramírez-Sánchez, and Francis M Tyers. 2011. Apertium: a free/open-source platform for rule-based machine translation. *Machine translation*, 25(2):127–144.

Arvi Hurskainen and Jörg Tiedemann. 2017. Rule-based machine translation from english to finnish. In *Proceedings of the Second Conference on Machine Translation*, pages 323–329.

Fred Karlsson. 1990. Constraint grammar as a framework for parsing unrestricted text. In *Proceedings of the 13th International Conference of Computational Linguistics*, volume 3, pages 168–173, Helsinki.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86.

Prasanth Kolachina and Aarne Ranta. 2015. Gf wide-coverage english-finnish mt system for wmt 2015. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 141–144.

Krister Lindn, Erik Axelson, Sam Hardwick, Tommi A Pirinen, and Miikka Silfverberg. 2011. Hfstframework for compiling and applying morphologies. *Systems and Frameworks for Computational Morphology*, pages 67–85.

Joss Moorkens, Antonio Toral, Sheila Castilho, and Andy Way. 2018. Translators perceptions of literary post-editing using statistical and neural machine translation. *Translation Spaces*, 7(2):240–262.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Tommi A Pirinen. 2015. 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.

Francis M Tyers, Felipe Sánchez-Martínez, Mikel L Forcada, et al. 2012. Flexible finite-state lexical selection for rule-based machine translation.

Weiyue Wang, Jan-Thorsten Peter, Hendrik Rosendahl, and Hermann Ney. 2016. Character: Translation edit rate on character level. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, volume 2, pages 505–510.