Apertium-IceNLP: A rule-based Icelandic to English machine translation system

Martha Dís Brandt, Hrafn Loftsson, Hlynur Sigurþórsson
School of Computer Science
Reykjavik University
IS-101 Reykjavik, Iceland
{marthab08,hrafn,hlynurs06}@ru.is

Francis M. Tyers
Dept. Lleng. i. Sist. Inform.
Universitat d'Alacant
E-03071 Alacant, Spain
ftyers@dlsi.ua.es

Abstract

We describe the development of a prototype of an open source rule-based Icelandic→English MT system, based on the Apertium MT framework and IceNLP, a natural language processing toolkit for Icelandic. Our system, *Apertium-IceNLP*, is the first system in which the whole morphological and tagging component of Apertium is replaced by modules from an external system. Evaluation shows that the word error rate and the position-independent word error rate for our prototype is 50.6% and 40.8%, respectively. As expected, this is higher than the corresponding error rates in two publicly available MT systems that we used for comparison. Contrary to our expectations, the error rates of our prototype is also higher than the error rates of a comparable system based solely on Apertium modules. Based on error analysis, we conclude that better translation quality may be achieved by replacing only the tagging component of Apertium with the corresponding module in IceNLP, but leaving morphological analysis to Apertium.

1 Introduction

Over the last decade or two, statistical machine translation (SMT) has gained significant momentum and success, both in academia and industry. SMT uses large parallel corpora, texts that are translations of each other, during training to derive a statistical translation model which is then used to translate between the source language (SL) and the target language (TL).

SMT has many advantages, e.g. it is data-driven, language independent, does not need linguistic experts, and prototypes of new systems can by built quickly and at a low cost. On the other hand, the need for parallel corpora as training data in SMT is also its main disadvantage, because such corpora are not available for a myriad of languages, especially the so-called *less-resourced languages*, i.e. languages for which few, if any, natural language processing (NLP) resources are available. When there is a lack of parallel corpora, other machine translation (MT) methods, such as rule-based MT, e.g. *Apertium* (Forcada et al., 2009), may be used to create MT systems.

In this paper, we describe the development of a prototype of an open source rule-based Icelandic→English (*is-en*) MT system based on Apertium and *IceNLP*, an NLP toolkit for processing and analysing Icelandic texts (Loftsson and Rögnvaldsson, 2007b). A decade ago, the Icelandic language could have been categorised as a less-resourced language. The current situation, however, is much better thanks to the development of IceNLP and various linguistic resources (Rögnvaldsson et al., 2009). On the other hand, no large parallel corpus, in which Icelandic is one of the languages, is freely available. This is the main reason why the work described here was initiated.

Our system, *Apertium-IceNLP*, is the first system in which the whole morphological and tagging component of Apertium is replaced by modules from an external system. Our motivation for developing such a *hybrid* system was to be able to answer the following research question: Is the translation quality of an *is-en* shallow-transfer MT system higher when using state-of-the-art Ice-

Mikel L. Forcada, Heidi Depraetere, Vincent Vandeghinste (eds.)
Proceedings of the 15th Conference of the European Association for Machine Translation, p. 217–224
Leuven, Belgium, May 2011

landic NLP modules in the Apertium pipeline as opposed to relying solely on Apertium modules?

Evaluation results show that the word error rate (WER) of our prototype is 50.6% and the position-independent word error rate (PER) is 40.8%[1]. This is higher than the evaluation results of two publicly available MT systems for *is-en* translation, *Google Translate*[2] and *Tungutorg*[3]. This was expected, given the short development time of our system, i.e. 8 man-months. For comparison, we know that Tungutorg has been developed by an individual, Stefán Briem, intermittently over a period of two decades[4].

Contrary to our expectations, the error rates of our hybrid system is also higher than the error rates of an *is-en* system based solely on Apertium modules. This "pure" Apertium version was developed in parallel with Apertium-IceNLP. Based on our error analysis, we conclude that better translation quality may be achieved by replacing only the tagging component of Apertium with the corresponding module in IceNLP, but leaving morphological analysis to Apertium.

We think that our work can be viewed as a guideline for other researchers wanting to develop hybrid MT systems based on Apertium.

2 Apertium

The Apertium shallow-transfer MT platform was originally aimed at the Romance languages of the Iberian peninsula, but has also been adapted for other languages, e.g. Welsh (Tyers and Donnelly, 2009) and Scandinavian languages (Nordfalk, 2009). The whole platform, both programs and data, is free and open source and all the software and data for the supported language pairs is available for download from the project website[5].

The Apertium platform consists of the following main modules:

- **A morphological analyser**: Performs tokenisation and morphological analysis which for a given surface form returns all of the possible lexical forms (analyses) of the word.
- **A part-of-speech (PoS) tagger**: The HMM-based PoS tagger, given a sequence of morphologically analysed words, chooses the most likely sequence of PoS tags.
- **Lexical selection**: A lexical selection module based on Constraint Grammar (Karlsson et al., 1995) selects between possible translations of a word based on sentence context.
- **Lexical transfer**: For an unambiguous lexical form in the SL, this module returns the equivalent TL form based on a bilingual dictionary.
- **Structural transfer**: Performs local morphological and syntactic changes to convert the SL into the TL.
- **A morphological generator**: For a given TL lexical form, this module returns the TL surface form.

2.1 Language pair specifics

For each language pair, the Apertium platform needs a monolingual SL dictionary used by the morphological analyser, a bilingual SL-TL dictionary used by the lexical transfer module, a monolingual TL dictionary used by the morphological generator, and transfer rules used by the structural transfer module. The dictionaries and transfer rules specific to the *is-en* pair will be discussed in Sections 4.2 and 4.3, respectively.

The lexical selection module is a new module in the Apertium platform and the *is-en* pair is the first released pair to make extensive use of it. The module works by selecting a translation based on sentence context. For example, for the ambiguous word *bóndi* 'farmer' or 'husband', the default translation is left as 'farmer', but a lexical selection rule chooses the translation of 'husband' if a possessive pronoun is modifying it. While the current lexical selection rules have been written by hand, work is ongoing to generate them automatically with machine learning techniques.

3 IceNLP

IceNLP is an open source[6] NLP toolkit for processing and analysing Icelandic texts. Currently, the main modules of IceNLP are the following:

- **A tokeniser.** This module performs both word tokenisation and sentence segmentation.

[1] See explanations of WER and PER in Section 5.
[2] http://translate.google.com
[3] http://www.tungutorg.is/
[4] We do not have information on development months for the *is-en* part of Google Translate.
[5] http://www.apertium.org
[6] http://icenlp.sourceforge.net

- ***IceMorphy*: A morphological analyser** (Loftsson, 2008). The program provides the tag profile (the ambiguity class) for known words by looking up words in its dictionary. The dictionary is derived from the *Icelandic Frequency Dictionary* (IFD) corpus (Pind et al., 1991). The tag profile for unknown words, i.e. words not known to the dictionary, is guessed by applying rules based on morphological suffixes and endings. IceMorphy does not generate word forms, it only carries out analysis.

- ***IceTagger*: A linguistic rule-based PoS tagger** (Loftsson, 2008). The tagger produces disambiguated morphosyntactic tags from the tagset of the IFD corpus. The tagger uses IceMorphy for morphological analysis and applies both local rules and heuristics for disambiguation.

- ***TriTagger*: A statistical PoS tagger**. This trigram tagger is a re-implemenation of the well-known HMM tagger described by Brants (2000). It is trained on the IFD corpus.

- ***Lemmald*: A lemmatiser** (Ingason et al., 2008). The method used combines a data-driven method with linguistic knowledge to maximise accuracy.

- ***IceParser*: A shallow parser** (Loftsson and Rögnvaldsson, 2007a). The parser marks both constituent structure and syntactic functions using a cascade of finite-state transducers.

3.1 The tagset and the tagging accuracy

The IFD corpus consists of about 600,000 tokens and the tagset of about 700 tags. In this tagset, each character in a tag has a particular function. The first character denotes the *word class*. For each word class there is a predefined number of additional characters (at most six), which describe morphological features, like *gender*, *number* and *case* for nouns; *degree* and *declension* for adjectives; *voice*, *mood* and *tense* for verbs, etc. To illustrate, consider the Icelandic word *strákarnir* 'the boys'. The corresponding IFD tag is *nkfng*, denoting noun (*n*), masculine (*k*), plural (*f*), nominative (*n*), and suffixed definite article (*g*).

Previous work on PoS tagging Icelandic text (Helgadóttir, 2005; Loftsson, 2008; Dredze and Wallenberg, 2008; Loftsson et al., 2009) has shown that the morphological complexity of the Icelandic language, and the relatively small training corpus in relation to the size of the tagset, is to blame for a rather low tagging accuracy (compared to related languages). Taggers that are purely based on machine learning (including HMM trigram taggers) have not been able to produce high accuracy when tagging Icelandic text (with the exception of Dredze and Wallenberg (2008)). The current state-of-the-art tagging accuracy of 92.5% is obtained by applying a hybrid approach, integrating TriTagger into IceTagger (Loftsson et al., 2009).

4 Apertium-IceNLP

We decided to experiment with using IceMorphy, Lemmald, IceTagger and IceParser in the Apertium pipeline. Note that since Apertium is based on a collection of modules that are connected by clean interfaces in a pipeline (following the Unix philosophy (Forcada et al., 2009)) it is relatively easy to replace modules or add new ones. Figure 1 shows the Apertium-IceNLP pipeline.

Our motivation for using the above modules is the following:

1. Developing a good morphological analyser for a language is a time-consuming task[7]. Since our system is unidirectional, i.e. *is-en* but not *en-is*, we only need to be able to analyse an Icelandic surface form, but do not need to generate an Icelandic surface form from a lexical form (lemma and morphosyntactic tags). We can thus rely on IceMorphy for morphological analysis.

2. As discussed in Section 3.1, research has shown that HMM taggers, like the one included in Apertium, have not been able to achieve high accuracy when tagging Icelandic. Thus, it seems logical to use the state-of-the-art tagger, IceTagger, instead.

3. Morphological analysers in Apertium return a lemma in addition to morphosyntactic tags. To produce a lemma for each word, we can instead rely on the Icelandic lemmatiser, Lemmald.

[7] Although there exists a morphological database for Icelandic (`http://bin.arnastofnun.is`), it is unfortunately not available as free/open source software/data.

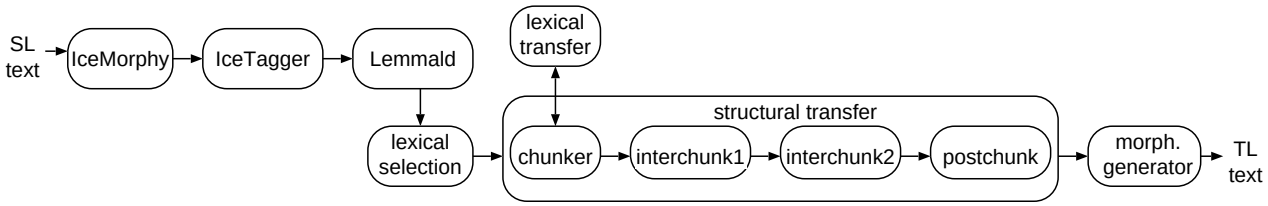

Figure 1: The Apertium-IceNLP pipeline.

4. Information about syntactic functions can be of help in the translation process. IceParser, which provides this information, can therefore potentially be used (we have not yet added IceParser to the pipeline).

4.1 IceNLP enhancements

In order to use modules from IceNLP in the Apertium pipeline, various enhancements needed to be carried out in IceNLP.

4.1.1 Mappings

Various mappings from the output generated by IceTagger to the format expected by the Apertium modules were necessary. All mappings were implemented by a single mapping file with different sections for different purposes. For example, morphosyntactic tags produced by IceTagger needed to be mapped to the tags used by Apertium. The mapping file thus contains entries for each possible tag from the IFD tagset and the corresponding Apertium tags. For example, the following entry in the mapping file shows the mapping for the IFD tag *nkfng* (see Section 3.1).

```
[TAGMAPPING]
...
nkfng <n><m><pl><nom><def>
```

The string "[TAGMAPPING]" above is a section name, whereas <n> stands for noun, <m> for masculine, <pl> for plural, <nom> for nominative, and <def> for definite.

Another example of a necessary mapping regards exceptions to tag mappings for particular lemmata. The following entries show that after tag mapping, the tags <vblex><actv> (verb, active voice) for the lemmata *vera* 'to be' and *hafa* 'to have' should be replaced by the single tag <vbser> and <vbhaver>, respectively. The reason is that Apertium needs specific tags for these verbs.

```
[LEMMA]
...
vera <vblex><actv> <vbser>
hafa <vblex><actv> <vbhaver>
```

The last example of a mapping concerns multiword expressions (MWEs). IceTagger tags each word of a MWE, whereas Apertium handles them as a single unit because MWEs cannot be translated word-for-word. Therefore, MWEs need to be listed in the mapping file along with the corresponding Apertium tags. The following entries show two MWEs, *að einhverju leyti* 'to some extent' and *af hverju* 'why' along with the corresponding Apertium tags.

```
[MWE]
...
að_einhverju_leyti <adv>
af_hverju <adv><itg>
```

Instead of producing tags for each component of a MWE, IceTagger searches for MWEs in its input text that match entries in the mapping file and produces the Apertium tag(s) for a particular MWE if a match is found.

4.1.2 Daemonising IceNLP

The versions of IceMorphy/IceTagger described in (Loftsson, 2008), and Lemmald described in (Ingason et al., 2008), were designed to tag and lemmatise large amounts of Icelandic text, e.g. corpora. When IceTagger starts up, it creates an instance of IceMorphy which in turn loads various dictionaries into memory. Similarily, when Lemmald starts up, it loads its rules into memory. This behaviour is fine when tagging and lemmatising corpora, because, in that case, the startup time is relatively small compared to the time needed to tag and lemmatise.

On the other hand, a common usage of a machine translation system is translating a small number of sentences (for example, in online MT services) as opposed to a corpus. Using the modules from IceNLP unmodified as part of the Apertium pipeline would be inefficient in that case because the aforementioned dictionaries and rules would be reloaded every time the language pair is used.

Therefore, we added a client-server functionality to IceNLP in order for it to run efficiently

as part of the Apertium pipeline. We added two new applications to IceNLP: *IceNLPServer* and *IceNLPClient*. IceNLPServer is a server application, which contains an instance of the IceNLP toolkit. Essentially, IceNLPServer is a daemon which runs in the background. When it is started up, all necessary dictionaries and rules are loaded into memory and are kept there while the daemon is running. Therefore, the daemon can serve requests to the modules in IceNLP without any loading delay.

IceNLPClient is a console-based client for communicating with IceNLPServer. This application behaves in the same manner as the Apertium modules, i.e. it reads from standard input and writes to standard output. Thus, we have replaced the Apertium tokeniser/morphological analyser/lemmatiser and the PoS tagger with IceNLPClient.

The client returns a PoS-tagged version of its input string. To illustrate, when the client is asked to analyse the string *Hún er góð* 'She is good':

```
echo "Hún er góð" | RunClient.sh
```

it returns:

```
^Hún/hún<prn><p3><f><sg><nom>$
^er/vera<vbser><pri><p3><sg>$
^góð/góður<adj><pst><f><sg><nom><sta>$
```

This output is consistent with the output generated by the Apertium tagger, i.e. for each word of an input sentence, the lexeme is followed by the lemma followed by the (disambiguated) morphosyntactic tags. The output above is then fed directly into the remainder of the Apertium pipeline, i.e. into lexical selection, lexical transfer, structural transfer, and morphological generation (see Figure 1), to produce the English translation 'She is good'.

4.2 The bilingual dictionary

When this project was initiated, no *is-en* bilingual dictionary (bidix) was publicly available in electronic format. Our bidix was built in three stages.

First, the *is-en* dictionary was populated with entries spidered from the Internet from Wikipedia, Wiktionary, Freelang, the Cleasby-Vigfusson Old Icelandic dictionary[8] and the Icelandic Word Bank[9]. This provided a starting point of over 5,000 entries in Apertium style XML format which needed to be checked manually for correctness. Also, since lexical selection was not an option in the early stages of the project, only one entry could be used. SL words that had multiple TL translations had to be commented out, based on which translation seemed the most likely option. For example, below we have three options for the SL word *fíngerður* in the bidix where it could be translated as 'fine', 'petite' or 'subtle', and the latter two options are commented out.

```
<e><p>
  <l>fíngerður<s n="adj"/></l>
  <r>fine<s n="adj"/><s n="sint"/></r>
</p></e>
<!-- begin comment
<e><p>
  <l>fíngerður<s n="adj"/></l>
  <r>petite<s n="adj"/></r>
</p></e>
<e><p>
  <l>fíngerður<s n="adj"/></l>
  <r>subtle<s n="adj"/></r>
</p></e>
end comment -->
```

Each entry in the bidix is surrounded by element tags `<e>...</e>` and paragraph tags `<p>...</p>`. SL words are surrounded by left tags `<l>...</l>` and TL translations by right tags `<r>...</r>`. Within the left and right tags the attribute value "adj" denotes that the word is an adjective and the presence of the attribute value "sint" denotes that the adjective's degree of comparison is shown with "-er/-est" endings (e.g. 'fine', 'finer', 'finest').

In the second stage of the bidix development, a bilingual wordlist of about 6,000 SL words with word class and gender was acquired from an individual, Anton Ingason. It required some preprocessing before it could be added to the bidix, e.g. determining which of these new SL words did not already exist in the dictionary, and selecting a default translation in cases where more than one translation was given.

Last, we acquired a bilingual wordlist from the dictionary publishing company *Forlagið*[10], containing an excerpt of about 18,000 SL words from their Icelandic-English online dictionary. This required similar preprocessing work as described above.

Currently, our *is-en* bidix contains 21,717 SL lemmata and 1,491 additional translations to be used for lexical selection.

4.3 Transfer rules

The syntactic (*structural*) transfer stage (see Figure 1) in the translator is split into four stages.

[8] http://www.ling.upenn.edu/~kurisuto/germanic/oi_cleasbyvigfusson_about.html

[9] http://www.ismal.hi.is/ob/index.en.html

[10] http://snara.is/

The first stage (*chunker*) performs local reordering and chunking. The second (*interchunk1*) produces chunks of chunks, e.g. chunking relative clauses into noun phrases. The third (*interchunk2*) performs longer distance reordering, e.g. constituent reordering, and some tense changes. As an example of a tense change, consider: *Hann vildi að verðlaunin færu til þeirra* → 'He wanted that the awards went to them' → 'He wanted the awards to go to them'. Finally, the fourth stage (*postchunk*) does some cleanup operations, and insertion of the indefinite article.

There are 78 rules in the first stage, the majority dealing with noun phrases, 3 rules in the second, 26 rules in the third stage and 5 rules in the fourth stage.

It is worth noting that the development of the bilingual dictionary and the transfer rules benefit both the Apertium-IceNLP system and the *is-en* system based solely on Apertium modules.

5 Evaluation

Our goal was to evaluate approximately 5,000 words, which corresponds to roughly 10 pages of text, and compare our results to two other publicly available *is-en* MT systems: Google Translate, an SMT system, and Tungutorg, a proprietary rule-based MT system, developed by an individual. In addition, we sought a comparison to the *is-en* system based solely on Apertium modules.

The test corpus for the evaluation was extracted from a dump of the Icelandic Wikipedia on April 24th 2010, which provided 187,906 lines of SL text. The reason for choosing texts from Wikipedia is that the evaluation material can be distributed, which is not the case for other available corpora of Icelandic.

Then, 1,000 lines were randomly selected from the test corpus and the resulting file filtered semi-automatically such that: *i)* each line had only one complete sentence; *ii)* each sentence had more than three words; *iii)* each sentence had zero or one lower case unknown word (we want to test the transfer, not the coverage of the dictionaries); *iv)* lines that were clearly metadata and traceable to individuals were removed, e.g. user names; *v)* lines that contained incoherent strings of numbers were removed, e.g. from a table entry; *vi)* lines containing non-Latin alphabet characters were removed, e.g. if they contained Greek or Arabic font; *vii)* lines that contained extremely domain specific and/or archaic words were removed (e.g. words that our human translator did not know how to translate); and *viii)* repetitive lines, e.g. multiple lines of the same format from a list, were removed.

Translator	WER	PER
Apertium-IceNLP	50.6%	40.8%
Apertium	45.9%	38.2%
Tungutorg	44.4%	33.7%
Google Translate	36.5%	28.7%

Table 1: Word error rate (WER) and position-independent word error rate (PER) over the test sentences for the publicly available *is-en* machine translation systems.

After this filtering process, 397 sentences remained which were then run through the four MT systems. In order to calculate evaluation metrics (see below), each of the four output files had to be post-edited. A bilingual human posteditor reviewed each TL sentence, copied it and then made minimal corrections to the copied sentence so that it would be suitable for dissemination – meaning that the sentence needs to be as close to grammatically correct as possible so that post-editing requires less effort.

The translation quality was measured using two metrics: word error rate (WER), and position-independent word error rate (PER). The WER is the percentage of the TL words that require correction, i.e. substitutions, deletions and insertions. PER is similar to WER except that PER does not penalise correct words in incorrect positions. Both metrics are based on the well known Levenshtein distance and were calculated for each of the sentences using the `apertium-eval-translator` tool[11]. Metrics based on word error rate were chosen so as to be able to compare the system against other Apertium systems and to assess the usefulness of the system in real settings, i.e. of translating for dissemination.

Note that, in our case, the WER and PER scores are computed based on the difference between the system output and a post-edited version of the system output. As can be seen in Table 1, the WER and PER for our Apertium-IceNLP prototype is 50.6% and 40.8%, respectively. This may seem quite high, but looking at the translation quality statistics for some of the other language pairs in Apertium[12], we see that

[11] `http://wiki.apertium.org/wiki/Evaluation`

[12] `http://wiki.apertium.org/wiki/`

the WER for Norsk Bokmål-Nynorsk is 17.7%, for Swedish-Danish 30.3%, for Breton-French 38.0%, for Welsh-English 55.7%, and for Basque-Spanish 72.4%. It is worth noting however that each of these evaluations had slightly different requirements for source language sentences. For instance, the Swedish–Danish pair allowed any number of unknown words.

We expected that the translation quality of Apertium-IceNLP would be significantly less than both Google Translate and Tungutorg, and the results in Table 1 confirm this expectation. The reason for our expectation was that the development time of our system was relatively short (8 man-months), whereas Tungutorg, for example, has been developed intermittently over a period of two decades.

Unexpectedly, the error rates of Apertium-IceNLP is also higher than the error rates of a system based solely on Apertium modules (see row "Apertium" in Table 1). We will discuss reasons for this and future work to improve the translation quality in the next section.

6 Discussion and future work

In order to determine where to concentrate efforts towards improving the translation quality of Apertium-IceNLP, some error analysis was carried out on a development data set. This development data was collected from the largest Icelandic online newspaper *mbl.is* into 1728 SL files and then translated by the system into TL files. Subsequently, 50 files from the pool were randomly selected for manual review and categorisation of errors.

The error categories were created along the way, resulting in a total of 6 error categories to identify where it would be most beneficial to make improvements. Analysis of the error categories showed that 60.7% of the errors were due to words missing from the bidix, mostly proper nouns and compound words (see Table 2). This analysis suggests that improvement to the translation quality can be achieved by concentrating on adding proper nouns to the bidix, on the one hand, and resolving compound words, on the other.

One possible explanation for the lower error rates for the "pure" Apertium version than the Apertium-IceNLP system is the handling of MWEs. MWEs most often do not translate literally nor even to the same number of words, which can dramatically increase the error rate. The pure version translates unlimited lengths of MWEs as single units and can deal with MWEs that contain inflectional words. In contrast, the length of the MWEs in IceNLP (and consequently also in Apertium-IceNLP) is limited to trigrams and, furthermore, IceNLP cannot deal with inflectional MWEs.

Error category	**Freq.**	**%**
Missing from the bidix	912	60.7%
Need further analysis	414	27.5%
Multiword expressions	90	6.0%
Abbreviations and initials	31	2.1%
More sophisticated patterns	31	2.1%
Other	24	1.6%
Total	**1502**	**100%**

Table 2: Error categories and corresponding frequencies.

The additional work required to get a better translation quality out of the Apertium-IceNLP system than a pure Apertium system raises the question as to whether "less is more", i.e. whether instead of incorporating tokenisation, morphological analysis, lemmatisation and PoS tagging from IceNLP into the Apertium pipeline, it may produce better results to only use IceTagger for PoS tagging but rely on Apertium for the other tasks. As discussed in Section 3.1, IceTagger outperforms an HMM tagger as the one used by the Apertium pipeline.

In order to replace only the PoS tagger in the Apertium pipeline, some modifications will have to be made to IceTagger. In addition to the modifications already carried out to make IceTagger return output in Apertium style format (see Section 4.1.1), the tagger will also have to be able to take Apertium style formatted input. More specifically, instead of relying on IceMorphy and Lemmald for morphological analysis and lemmatisation, IceTagger would have to be changed to receive the necessary information from the morphological component of Apertium.

7 Conclusion

We have described the development of Apertium-IceNLP, an Icelandic→English (*is-en*) MT system based on the Apertium platform and IceNLP, an NLP toolkit for Icelandic. Apertium-IceNLP is a hybrid system, the first system in which the whole morphological and tagging component of Aper-

`Translation_quality_statistics`

tium is replaced by modules from an external system.

Our system is a prototype with about 8 man-months of development work. Evaluation, based on word error rate, shows that our prototype does not perform as well as two other available *is-en* systems, Google Translate and Tungutorg. This was expected and can mainly be explained by two factors. First, our system has been developed over a short time. Second, our system makes systematic errors that we intend to fix in future work.

Contrary to our expectations, the Apertium-IceNLP system also performs worse than the *is-en* system based solely on Apertium modules. We conjectured that this is mainly due to the fact that the Apertium-IceNLP system does not handle MWEs adequately, whereas the handling of MWEs is an integrated part of the Apertium morphological analyser. Therefore, we expect that better translation quality may be achieved by replacing only the tagging component of Apertium with the corresponding module in IceNLP, but leaving morphological analysis to Apertium. This conjecture will be verified in future work.

Acknowledgments

The work described in this paper has been supported by: i) The Icelandic Research Fund, project "Viable Language Technology beyond English – Icelandic as a test case", grant no. 090662012; and ii) The NILS mobility project (The Abel Predoc Research Grant), coordinated by Universidad Complutense de Madrid.

References

Brants, Thorsten. 2000. TnT: A statistical part-of-speech tagger. In *Proceedings of the* 6^{th} *Conference on Applied Natural Language Processing*, Seattle, WA, USA.

Dredze, Mark and Joel Wallenberg. 2008. Icelandic Data Driven Part of Speech Tagging. In *Proceedings of the* 46^{th} *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Columbus, OH, USA.

Forcada, Mikel L., Francis M. Tyers, and Gema Ramírez-Sánches. 2009. The Apertium machine translation platform: Five years on. In *Proceedings of the First International Workshop on Free/Open-Source Rule-Based Machine Translation*, Alacant, Spain.

Helgadóttir, Sigrún. 2005. Testing Data-Driven Learning Algorithms for PoS Tagging of Icelandic. In Holmboe, H., editor, *Nordisk Sprogteknologi 2004*, pages 257–265. Museum Tusculanums Forlag, Copenhagen.

Ingason, Anton K., Sigrún Helgadóttir, Hrafn Loftsson, and Eiríkur Rögnvaldsson. 2008. A Mixed Method Lemmatization Algorithm Using Hierachy of Linguistic Identities (HOLI). In Nordström, B. and A. Rante, editors, *Advances in Natural Language Processing,* 6^{th} *International Conference on NLP, GoTAL 2008, Proceedings*, Gothenburg, Sweden.

Karlsson, Fred, Atro Voutilainen, Juha Heikkilä, and Arto Anttila. 1995. *Constraint Grammar: A Language-Independent System for Parsing Unrestricted Text*. Mouton de Gruyter, Berlin.

Loftsson, Hrafn and Eiríkur Rögnvaldsson. 2007a. IceParser: An Incremental Finite-State Parser for Icelandic. In *Proceedings of the* 16^{th} *Nordic Conference of Computational Linguistics (NoDaLiDa 2007)*, Tartu, Estonia.

Loftsson, Hrafn and Eiríkur Rögnvaldsson. 2007b. IceNLP: A Natural Language Processing Toolkit for Icelandic. In *Proceedings of Interspeech 2007, Special Session: "Speech and language technology for less-resourced languages"*, Antwerp, Belgium.

Loftsson, Hrafn, Ida Kramarczyk, Sigrún Helgadóttir, and Eiríkur Rögnvaldsson. 2009. Improving the PoS tagging accuracy of Icelandic text. In *Proceedings of the* 17^{th} *Nordic Conference of Computational Linguistics (NoDaLiDa 2009)*, Odense, Denmark.

Loftsson, Hrafn. 2008. Tagging Icelandic text: A linguistic rule-based approach. *Nordic Journal of Linguistics*, 31(1):47–72.

Nordfalk, Jacob. 2009. Shallow-transfer rule-based machine translation for Swedish to Danish. In *Proceedings of the First International Workshop on Free/Open-Source Rule-Based Machine Translation*, Alacant, Spain.

Pind, Jörgen, Friðrik Magnússon, and Stefán Briem. 1991. *Íslensk orðtíðnibók [The Icelandic Frequency Dictionary]*. The Institute of Lexicography, University of Iceland, Reykjavik.

Rögnvaldsson, Eiríkur, Hrafn Loftsson, Kristín Bjarnadóttir, Sigrún Helgadóttir, Anna B. Nikulásdóttir, Matthew Whelpton, and Anton K. Ingason. 2009. Icelandic Language Resources and Technology: Status and Prospects. In Domeij, R., K. Koskenniemi, S. Krauwer, B. Maegaard, E. Rögnvaldsson, and K. de Smedt, editors, *Proceedings of the NoDaLiDa 2009 Workshop 'Nordic Perspectives on the CLARIN Infrastructure of Language Resources'*. Odense, Denmark.

Tyers, Francis M. and Kevin Donnelly. 2009. apertium-cy - a collaboratively-developed free RBMT system for Welsh to English. *Prague Bulletin of Mathematical Linguistics*, 91:57–66.