Abu-MaTran at WMT 2015 Translation Task:
Morphological Segmentation and Web Crawling

Raphael Rubino⋆, Tommi Pirinen†, Miquel Esplà-Gomis‡, Nikola Ljubešićγ, Sergio Ortiz-Rojas⋆, Vassilis Papavassiliou‡, Prokopis Prokopidisγ, Antonio Toral†

⋆ Prompsit Language Engineering, S.L., Elche, Spain
{rrubino,sortiz}@prompsit.com
† NCLT, School of Computing, Dublin City University, Ireland
{atoral,tpirinen}@computing.dcu.ie
‡ Dep. Llenguatges i Sistemes Informàtics, Universitat d’Alacant, Spain
mespla@dlsi.ua.es
γ Department of Information and Communication Sciences, University of Zagreb, Croatia
nljubesi@ffzg.hr
♭ Institute for Language and Speech Processing, Athena Research and Innovation Center, Greece
{vpapa, prokopis}@ilsp.gr

Abstract

This paper presents the machine translation systems submitted by the Abu-MaTran project for the Finnish–English language pair at the WMT 2015 translation task. We tackle the lack of resources and complex morphology of the Finnish language by (i) crawling parallel and monolingual data from the Web and (ii) applying rule-based and unsupervised methods for morphological segmentation. Several statistical machine translation approaches are evaluated and then combined to obtain our final submissions, which are the top performing English-to-Finnish unconstrained (all automatic metrics) and constrained (BLEU), and Finnish-to-English constrained (TER) systems.

1 Introduction

This paper presents the statistical machine translation (SMT) systems submitted by the Abu-MaTran project for the WMT 2015 translation task. The language pair concerned is Finnish–English with a strong focus on the English-to-Finnish direction. The Finnish language is newly introduced this year as a particular translation challenge due to its rich morphology and to the lack of resources available, compared to e.g. English or French.

Morphologically rich languages, and especially Finnish, are known to be difficult to translate using phrase-based SMT systems mainly because of the large diversity of word forms leading to data scarcity (Koehn, 2005). We assume that data acquisition and morphological segmentation should contribute to decrease the out-of-vocabulary rate and thus improve the performance of SMT. To gather additional data, we decide to build on previous work conducted in the Abu-MaTran project and crawl the Web looking for monolingual and parallel corpora (Toral et al., 2014). In addition, morphological segmentation of Finnish is used in our systems as pre- and post-processing steps. Four segmentation methods are proposed in this paper, two unsupervised and two rule-based.

Both constrained and unconstrained translation systems are submitted for the shared task. The former ones are trained on the data provided by the shared task, while the latter ones benefit from crawled data. For both settings, we evaluate the impact of the different SMT approaches and morphological segmentation methods. Finally, the outputs of individually trained systems are combined to obtain our primary submissions for the translation tasks.

This paper is structured as follows: the methods for data acquisition from the Web are described in Section 2. Morphological segmentation is presented in Section 3. The data and tools used in our experiments are detailed in Section 4. Finally, the results of our experiments are shown in Section 5, followed by a conclusion in Section 6.

2 Web Crawling

In this section we describe the process we followed to collect monolingual and parallel data through Web crawling. Both types of corpora are gathered through one web crawl of the Finnish (.fi) top-level
domain (TLD) with the SPIDERLING crawler\(^1\) (Suchomel and Pomikálek, 2012). This crawler performs language identification during the crawling process and thus allows simultaneous multilingual crawling. The whole unconstrained dataset gathered from the Web is built in 40 days using 16 threads. Documents written in Finnish and English are collected during the crawl.

### 2.1 Monolingual Data

The Finnish and English data collected during the crawl amounts to 5.6\(M\) and 3.9\(M\) documents, containing 1.7\(B\) and 2.0\(B\) words for Finnish and English respectively (after processing, which includes removing near-duplicates). Interestingly, the amount of Finnish and English data on the Finnish TLD is quite similar. For comparison, on the Croatian domain only 10% of the data is written in English (Ljubešić and Klubička, 2014). While the Finnish data is used in further steps for building the target-language model, both datasets are used in the task of searching for parallel data described in the next subsection.

### 2.2 Parallel Data

In our experiments, we adapt the BITEXTOR\(^2\) tool to detect parallel documents from a collection of downloaded and pre-processed websites. The pre-processing performed by SPIDERLING includes language detection, boilerplate removal, and HTML format cleaning. Therefore, the only modules of BITEXTOR used for this task are those performing document and segment alignment, relying on HUNALIGN\(^3\) (Varga et al., 2005) and an English–Finnish bilingual lexicon.\(^4\) Confidence scores for aligned segments are computed thanks to these two resources.

From a total of 12.2\(K\) web domains containing both Finnish and English documents, BITEXTOR is able to identify possible parallel data on 10.7\(k\) domains (87.5%). From these domains, 2.1\(M\) segment pairs are extracted without any additional restrictions, and 1.2\(M\) when additional restrictions on the document pairing are set. Namely, these restrictions discard (i) document pairs where less than 5 segments are aligned; and (ii) those with an alignment score lower than 0.2 according to HUNALIGN. The first collection can be considered recall-oriented and the second one precision-oriented.

In this first step, a large amount of potentially parallel data is obtained by post-processing data collected with a TLD crawl, which is not primarily aimed at finding parallel data. To make use of this resource in a more efficient way, we re-crawl some of the most promising web sites (we call them multilingual hotspots) with the ILSP-FC crawler specialised in locating parallel documents during crawling. According to Esplà-Gomis et al. (2014), BITEXTOR and ILSP-FC have shown to be complementary, and combining both tools leads to a larger amount of parallel data.

ILSP-FC (Papavassiliou et al., 2013) is a modular crawling system allowing to easily acquire domain-specific and generic corpora from the Web.\(^5\) This crawler includes a de-duplicator which checks all documents in a pairwise manner to identify near-duplicates. This is achieved by comparing the quantised word frequencies and the paragraphs of each pair of candidate duplicate documents. A document-pair detector also examines each document in the same manner and identifies pairs of documents that could be considered parallel. The main methods used by the pair detector are URL similarity, co-occurrences of images with the same filename in two documents, and the documents’ structural similarity.

In order to identify the multilingual hotspots, we process the output of the Finnish TLD and generate a list containing the websites which have already been crawled and the number of stored English and Finnish webpages for each website. Assuming that a website with comparable numbers of webpages for each language is likely to contain bitexts of good quality, we keep the websites with Finnish to English ratio over 0.9. Then, ILSP-FC processes the 1, 000 largest such websites, considered the most bitext-productive multilingual websites, in order to detect parallel documents. We identify a total of 58, 839 document pairs (8, 936, 17, 288 and 32, 615 based on URL similarity, co-occurrences of images and structural similarity, respectively). Finally, HUNALIGN is applied on these document pairs, resulting in 1.2\(M\) segment pairs after duplicate removal. The parallel corpus used in our experiments is the union without duplicates of the largest

\(^1\)http://nlp.fi.muni.cz/trac/spiderling
\(^2\)http://sf.net/p/bitextor/
\(^3\)http://mokk.bme.hu/resources/hunalign
\(^4\)http://sf.net/p/bitextor/files/bitextor/bitextor-4.1/dictionaries/
\(^5\)http://nlp.ilsp.gr/redmine/projects/ilsp-fc
corpora collected with BITEXTOR and ILSP-FC, leading to 2.8M segment pairs.

3 Morphological Segmentation

Morphological segmentation is a method of analysis of word-forms in order to reduce morphological complexity. There are few variations on how to define morphological segmentation, we use the most simple definition: a morphological segmentation of a word is defined by 0 or more segmentation points from where the word can be split into segments. The letter sequences between segmentation points are not modified, i.e. no lemmatisation or segment analysis is performed (or retained) in the actual SMT data. An example of a linguistically derived morphological segmentation of an English word-form cats would be cat→←s, where →← denotes the segmentation point,\(^6\) and cat and s are the segments.

We use four segmentation approaches that can be divided in two categories: (i) rule-based, based on morphological dictionaries and weighted finite-state technology HFST (Lindén et al., 2009)\(^7\), further detailed in subsection 3.1, and (ii) statistical, based on unsupervised learning of morphologies, further detailed in subsection 3.2. All segments are used as described in subsection 3.3.

3.1 Rule-based Segmentation

Rule-based morphological segmentation is based on linguistically motivated computational descriptions of the morphology by dividing the word-forms into morphs (minimal segments carrying semantic or syntactic meaning). The rule-based approach to morphological segmentation uses a morphological dictionary of words and an implementation of the morphological grammar to analyse word-forms. In our case, we use OMORFI (Pirinen, 2015), an open-source implementation of the Finnish morphology.\(^8\) OMORFI’s segmentation produces named segment boundaries: stem, inflection, derivation, compound-word and other etymological. The two variants of rule-based segmentation we use are based on selection of the boundary points: compound segmentation uses compound segments and discards the rest (referred in tables and figures to as HFST Comp), and morph segmentation uses compound and inflectional morph segments (HFST Morph in tables and figures). In cases of ambiguous segments, the weighted finite-state automata 1-best search is used with default weights.\(^9\) For example, the words kuntaliitoksen selvittämisessä (“examining annexation”) is segmented by hfst-comp as ‘kunta→←liitoksen selvittämisessä’ and hfst-morph as ‘kunta→←liitokse→←n selvittämis→←sä’.

3.2 Unsupervised Segmentation

Unsupervised morphological segmentation is based on a statistical model trained by minimising the number of different character sequences observed in a training corpus. We use two different algorithms: MORFESSOR Baseline 2.0 (Virpioja et al., 2013) and FLATCAT (Grönroos et al., 2014). The segmentation models are trained using the Europarl v8 corpus. Both systems are used with default settings. However, with FLATCAT we discard the non-morph boundaries and we have not used semi-supervised features. For example, the phrase given in previous sub-section: morfessor produces 1-best segmentation: and ‘Kun→←ta→←liito→←ksen selvittä→←misessä’ and flatcat ‘Kun→←tali→←itokse selvittämis→←essä’

3.3 Segments in the SMT Pipeline

The segmented data is used exactly as the word-form-based data during training, tuning and testing of the SMT systems,\(^10\) except during the pre-processing and post-processing steps. For pre-processing, the Finnish side is segmented prior to use. For the post-processing of segmented-Finnish-to-English, boundary markers are removed. For the other direction, two types of tokens with boundary markers are observed: matching arrows a→←b and stray arrows a→ x or x ← b. For matching arrows, an empty string is used to join the morphs, while the morphs with stray arrows are deleted.

4 Datasets and Tools

This section presents the tools, the monolingual and parallel data used to train our SMT systems. All the corpora are pre-processed prior to training the

\(^6\)we follow this arrow notation throughout the paper as well as in the actual implementation

\(^7\)http://hfst.sf.net

\(^8\)http://github.com/flammie/omorfi/

\(^9\)For details of implementation and reproducibility, the code is available in form of automake scriptlets at http://github.com/flammie/autostuff-moses-smt./

\(^10\)The parameters of the word alignment, phrase extraction and decoding algorithms have not been modified to take into account the nature of the segmented data.
language and translation models. We rely on the scripts included in the MOSES toolkit (Koehn et al., 2007) and perform the following operations: punctuation normalisation, tokenisation, true-casing and escaping of problematic characters. The truecaser is lexicon-based, trained on all the monolingual and parallel data. In addition, we remove sentence pairs from the parallel corpora where either side is longer than 80 tokens.

4.1 Translation Models

We empirically evaluate several types of SMT systems: phrase-based SMT (Och and Ney, 2004) trained on word forms or morphs as described in Section 3, Factored Models (Koehn and Hoang, 2007) including morphological and suffix information as provided by OMORFI,\footnote{Using the script omorfifactorise.py} in addition to surface forms, and finally hierarchical phrase-based SMT (Chiang, 2005) as an unsupervised tree-based model. All the systems are trained with MOSES, relying on MGIZA (Gao and Vogel, 2008) for word alignment and MIRA (Watanabe et al., 2007) for tuning. This tuning algorithm was shown to be faster and as efficient as MERT for model core features, as well as a better stability with larger numbers of features (Hasler et al., 2011).

In order to compare the individually trained SMT systems, we use the same parallel data for each model, as well as the provided development set to tune the systems. The phrase-based SMT system is augmented with additional features: an Operation Sequence Model (OSM) (Durrani et al., 2011) and a Bilingual Neural Language Model (BiNLM) (Devlin et al., 2014), both trained on the parallel data used to learn the phrase-table. All the translation systems also benefit from two additional reordering models, namely a phrase-based model with three different orientations (monotone, swap and discontinuous) and a hierarchical model with four orientations (non merged discontinuous left and right orientations), both trained in a bidirectional way (Koehn et al., 2005; Galley and Manning, 2008).

Our constrained systems are trained on the data available for the shared task, while unconstrained systems are trained with two additional sets of parallel data, the FiENWAC crawled dataset (cf. Section 2.2) and Open Subtitles, henceforth OSUBS.\footnote{http://opus.lingfil.uu.se/}

The details about the corpora used to train the translation models are presented in Table 1. Figure 1 shows how different segmentation methods affect the vocabulary size; given that linguistic segmentation have larger vocabularies as statistical their contribution to translation models may be at least partially complementary.

The two unconstrained parallel datasets are split into three subsets: pseudo in-domain, pseudo out-of-domain top and pseudo out-of-domain bottom, henceforth in, outt and outb. We rank the sentence pairs according to bilingual cross-entropy difference on the devset (Axelrod et al., 2011) and calculate the perplexity on the devset of LMs trained on different portions of the top ranked sentences (the top 1/64, 1/32 and so on). The subset for which we obtain the lowest perplexities is kept as in (this was 1/4 for fienwac (403.89 and 3610.95 for English and Finnish, respectively), and 1/16 for osubs (702.45 and 7032.2). The remaining part of each dataset is split in two sequential parts in ranking order of same number of lines, which are kept as outt and outb.

The out-of-domain part of osubs is further processed with vocabulary saturation (Lewis and Eteemadi, 2013) in order to have a more efficient and compact system (Rubino et al., 2014). We traverse the sentence pairs in the order they are ranked.

![Figure 1: Effects of segmentation on unique token counts for Finnish.](image)

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Sentences (k)</th>
<th>Words (M)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Finnish</td>
<td>English</td>
</tr>
<tr>
<td>Europarl</td>
<td></td>
<td></td>
</tr>
<tr>
<td>v8</td>
<td>1.901.1</td>
<td>36.5</td>
</tr>
<tr>
<td></td>
<td><strong>Constrained</strong></td>
<td><strong>Unconstrained</strong></td>
</tr>
<tr>
<td>fienus. in</td>
<td>640.1</td>
<td>9.2</td>
</tr>
<tr>
<td>fienus. outt</td>
<td>838.9</td>
<td>12.5</td>
</tr>
<tr>
<td>fienus. outb</td>
<td>838.9</td>
<td>13.9</td>
</tr>
<tr>
<td>osubs.in</td>
<td>492.2</td>
<td>3.6</td>
</tr>
<tr>
<td>osubs.outt</td>
<td>1,169.6</td>
<td>8.8</td>
</tr>
<tr>
<td>osubs.outb</td>
<td>1,169.6</td>
<td>7.8</td>
</tr>
</tbody>
</table>

Table 1: Parallel data used to train the translation models, after pre-processing.
Table 2: English monolingual data, after pre-processing, used to train the constrained language model.

Table 3: Finnish monolingual data, after pre-processing, used to train the language models.

Table 4: Results obtained on the development and test sets for the constrained English-to-Finnish translation task. Best individual system in bold.

4.2 Language Models

All the Language Models (LM) used in our experiments are 5-grams modified Kneser-Ney smoothed LMs trained using KenLM (Heafield et al., 2013). For the constrained setup, the Finnish and the English LMs are trained following two different approaches. The English LM is trained on the concatenation of all available corpora while the Finnish LM is obtained by linearly interpolating individually trained LMs based on each corpus. The weights given to each individual LM is calculated by minimising the perplexity obtained on the development set. For the unconstrained setup, the Finnish LM is trained on the concatenation of all constrained data plus the additional monolingual crawled corpora (noted FiWaC). The data used to train the English and Finnish LMs are presented in Table 2 and Table 3 respectively.

5 Results

We tackle the English-to-Finnish direction in the unconstrained task, while both directions are presented for the constrained task. Systems’ outputs are combined using MEMT (Heafield and Lavie, 2010) using default settings, except for the beam size (set to 1, 500) and radius (5 for Finnish and 7 for English), following empirical results obtained on the development set.

5.1 Constrained Results

Individual systems trained on the provided data are evaluated before being combined. The results obtained for the English-to-Finnish direction are presented in Table 4. The BLEU (Papineni et al., 2002) and TER (Snover et al., 2006) scores obtained by the system trained on compound-segmented data (HFST Comp) show a positive impact of this method on SMT according to the development set, compared to the other individual systems. The unsupervised segmentation methods do not improve over phrase-based SMT, while the hierarchical model shows an interesting reduction of the TER score compared to a classic phrase-based approach. On the test set, the use of inflectional morph segments as well as compounds (HFST Morph) leads to the best results for the individual systems on both evaluation metrics. The combination of these 7 systems improves substantially over the best individual system for the development and the test sets.

The results for the other translation direction (Finnish to English) are shown in Table 5 and

13We use NIST mteval v13 and TERp v0.1, both with default parameters.
Table 5: Results obtained on the development and test sets for the constrained Finnish-to-English translation task. Best individual system in bold.

Table 6: Results obtained on the development and test sets for the unconstrained English-to-Finnish translation task. Best individual system in bold.

follow the same trend as observed with Finnish as target: the morphologically segmented data helps improving over classic SMT approaches.

The two metrics indicate better performances of HFST Morph on the development set, while Flatcat reaches the best scores on the test set. The results obtained with the segmented data on the two translation directions and the different segmentation approaches are fluctuating and do not indicate which method is the best. Again, the combination of all the systems results in a substantial improvement over the best individual system across both evaluation metrics. The top 3 systems presented in Table 5, namely Combination, HFST Morph and Phrase-Based correlates with the results reported by the manual evaluation.\footnote{http://www.statmt.org/wmt15/results.html}

5.2 Unconstrained Results

We present the results obtained on the unconstrained English-to-Finnish translation task in Table 6. Two individual systems are evaluated, using word-forms and compound-based data, and show that the segmented data leads to lower TER scores, while higher BLEU are reached by the word-based system. The combination of these two systems in addition to the constrained outputs of the remaining systems (hierarchical, factored model, HFST Morph, Morfessor and Flatcat) is evaluated in the last row of the table, and shows .3pt BLEU gain on the test set over the phrase-based approach using word forms.

The human evaluation conducted on the English–Finnish translation direction shows interesting results. While our unconstrained Combination system outperforms our other manually evaluated systems, the quality of the unconstrained Phrase-Based output is lower than the constrained Combination one. The opposite is observed on the automatic metrics, with a difference of 2.5pts BLEU and .2pt TER.

6 Conclusion

Our participation in WMT15’s translation task focuses on investigating the use of several morphological segmentation methods and Web data acquisition in order to handle the data scarcity and the rich morphology of Finnish. We evaluate several SMT approaches, showing the usefulness of morphological segmentation for Finnish SMT. In particular, the rule-based methods lead to the best results on the constrained English–Finnish task compared to our other individual systems.

In addition, the manual evaluation results indicate that combining diverse SMT systems’ outputs, including morphologically segmented ones, can outperform a classic phrase-based approach trained on larger parallel and monolingual corpora. The combination of the different SMT systems leads to the best results for both translation directions, as shown by automatic metrics and manual evaluation. Finally, the acquisition of additional training data improves over the constrained systems and is a successful example of the Abu-MaTran crawling pipeline. However, the discrepancy observed on the results using the different segmentation methods requires a deeper analysis of the SMT output, which is planned as future work.

Acknowledgments

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement PIAP-GA-2012-324414 (Abu-MaTran). We would like to thank Kenneth Heafield for his help to our questions re MEMT.
References


