Edinburgh Research Explorer

The Edinburgh/JHU Phrase-based Machine Translation Systems for WMT 2015

Citation for published version:
<http://aclweb.org/anthology/W15-3013>

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Publisher's PDF, also known as Version of record

Published In:
Proceedings of the Tenth Workshop on Statistical Machine Translation

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
The Edinburgh/JHU Phrase-based Machine Translation Systems for WMT 2015

Barry Haddow\textsuperscript{1}, Matthias Huck\textsuperscript{1}, Alexandra Birch\textsuperscript{1}, Nikolay Bogoychev\textsuperscript{1}, Philipp Koehn\textsuperscript{1,2}

\textsuperscript{1}School of Informatics, University of Edinburgh
\textsuperscript{2}Center for Speech and Language Processing, The Johns Hopkins University

a.birch@ed.ac.uk \{nbogoych,bhaddow,mhuck\}@inf.ed.ac.uk phi@jhu.edu

Abstract

This paper describes the submission of the University of Edinburgh and the Johns Hopkins University for the shared translation task of the EMNLP 2015 Tenth Workshop on Statistical Machine Translation (WMT 2015). We set up phrase-based statistical machine translation systems for all ten language pairs of this year’s evaluation campaign, which are English paired with Czech, Finnish, French, German, and Russian in both translation directions.

Novel research directions we investigated include: neural network language models and bilingual neural network language models, a comprehensive use of word classes, and sparse lexicalized reordering features.

1 Introduction

The Edinburgh/JHU phrase-based translation systems for our participation in the WMT 2015 shared translation task\textsuperscript{1} are based on the open source Moses toolkit (Koehn et al., 2007). We built upon Edinburgh’s strong baselines from WMT submissions in previous years (Durrani et al., 2014a) as well as our recent research within the framework of other evaluation campaigns and projects such as IWSLT\textsuperscript{2} and EU-BRIDGE\textsuperscript{3} (Birch et al., 2014; Freitag et al., 2014a; Freitag et al., 2014b).

We first discuss novel features that we integrated into our systems for the 2015 Edinburgh/JHU submission. Next we give a general system overview with details on our training pipeline and decoder configuration. We finally present empirical results for the individual language pairs and translation directions.

2 Novel Methods

2.1 Neural Network LM with NPLM

For some language pairs (notably French→English and Finnish→English) we experimented with feed-forward neural network language models using the NPLM toolkit (Vaswani et al., 2013). This toolkit enables such language models to be trained efficiently on large datasets, and provides a querying API which is fast enough to be used during decoding. NPLM is fully integrated into Moses, including appropriate wrapper scripts for training the language models within the Moses experiment management system.

2.2 Bilingual Neural Network LM

We also experimented with our re-implementation of the “joint” model by Devlin et al. (2014). Referred to as bilingual LM in Moses, this was previously employed in the Edinburgh IWSLT system submissions, although with limited success (Birch et al., 2014).

The idea of the bilingual LM is quite straightforward. We define a language model where each target token is conditioned on the previous \((n−1)\) target tokens (as in a standard \(n\)-gram language model) as well as its aligned source token, and a window of \(m\) tokens on either side of the aligned source token. At training time, the aligned source token is found from the automatic alignment, and at test time the alignment is supplied by the decoder. The bilingual LM is trained using a feed-forward neural network and we use the NPLM toolkit for this.

Prior to submission we tested bilingual LMs on the French→English tasks and on English→Russian task. For French→English, we had resource issues\textsuperscript{4} in training such large

\textsuperscript{1}http://www.statmt.org/wmt15/
\textsuperscript{2}http://workshop2014.iwslt.org
\textsuperscript{3}http://www.eu-bridge.eu
\textsuperscript{4}These can now be addressed using the -mmap option to create a binarized version of the corpus which is then memory-mapped.
models so we randomly subsampled 10% of the data for training. Since we did not observe gains in translation quality, the bilingual LM was not integrated into our primary system submissions. In post-submission experiments, we tried training bilingual LM on a 10% domain-specific portion of the training data selected using modified Moore-Lewis (Moore and Lewis, 2010; Axelrod et al., 2011), but only observed a small improvement in translation performance.

2.3 Comprehensive Use of Word Classes
In Edinburgh’s submission from the previous year, we used automatically generated word classes in additional language models and in additional operation sequence models (Durrani et al., 2014b). This year, we pushed the use of word classes into the remaining feature functions: the reordering model and the sparse word features.

We generated Och clusters (Och, 1999) — a variant of Brown clusters — using mkcls. We have to choose a hyper parameter: the number of clusters. Our experiments and also prior work (Stewart et al., 2014) suggest that instead of committing to a single value, it is beneficial to use multiple numbers and use them in multiple feature functions concurrently. We used 50, 200, 600, and 2000 clusters, hence having 4 additional interpolated language models, 4 additional operation sequence models, 4 additional lexicalized reordering models, and 4 additional sets of sparse features.

The feature functions for word classes were trained exactly the same way as the corresponding feature functions for words. For instance, this means that the word class language model required training of individual models on the subcorpora, and then interpolation.

We carried out a study to assess the contribution of the use of such word class feature functions. Table 1 summarizes the results. Use of word classes in each of the models yields small gains, except for the reordering model, where there is no observable difference. The biggest gains were observed in the language model. Note that the English–German baseline already included additional feature functions based on POS and morphological tags, and basically no additional gains were observed due to the class based feature functions.

2.4 Sparse Lexicalized Reordering
We implemented sparse lexicalized reordering features (Cherry, 2013) in Moses and evaluated them in English–German setups. The experiments were conducted on top of the standard hierarchical lexicalized reordering model (Galley and Manning, 2008). We applied features based on Och clusters with 200 classes on both source and target side. Active feature groups are between, phrase, and stack.

In addition to optimizing the feature weights directly with k-best MIRA (Cherry and Foster, 2012), we also examined maximum expected BLEU training of the sparse lexicalized reordering features via stochastic gradient descent (Auli et al., 2014).

3 System Overview
3.1 Preprocessing
The training data was preprocessed using scripts from the Moses toolkit. We first normalized the data using the normalize-punctuation.perl script, then performed tokenization (using the -a option), and then truecasing. We did not perform any corpus filtering other than the standard Moses method, which removes sentence pairs with extreme length ratios.

3.2 Word Alignment
For word alignment we used either fast_align (Dyer et al., 2013) or MGIZA++ (Gao and Vogel, 2008), followed by the standard grow-diag-final-and symmetrization heuristic. An empirical comparison of fast_align and MGIZA++ on the Finnish-English and English-Russian language pairs using the constrained data sets did not reveal any significant difference.

3.3 Language Model
We used all available monolingual data to train 5-gram language models with modified Kneser-Ney smoothing (Chen and Goodman, 1998). Typically, language models for each monolingual corpus were first trained using either KenLM (Heafield et al., 2013) or the SRILM toolkit (Stolcke, 2002) and then linearly interpolated using weights tuned to minimize perplexity on the development set.

3.4 Baseline Features
We follow the standard approach to SMT of scoring translation hypotheses using a weighted linear combination of features. The core features of our
Table 1: Use of additional feature functions based on Och clusters (see Section 2.3). The last four lines refer to ablation studies where one of the sets of clustered feature functions is removed from the comprehensive setup. Note that the word-based feature functions are used in all cases. BLEU scores on newstest2014 are reported.

<table>
<thead>
<tr>
<th></th>
<th>de-en</th>
<th>en-de</th>
<th>cs-en</th>
<th>en-cs</th>
<th>ru-en</th>
<th>en-ru</th>
<th>avg</th>
<th>∆</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline (no clusters)</td>
<td>28.0</td>
<td>20.5</td>
<td>29.1</td>
<td>21.2</td>
<td>31.8</td>
<td>29.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comprehensive setup</td>
<td>28.5 (+.5)</td>
<td>20.3 (+.2)</td>
<td>29.7 (+.6)</td>
<td>21.8 (+.6)</td>
<td>32.3 (+.5)</td>
<td>29.7 (+.6)</td>
<td></td>
<td>+.5</td>
</tr>
<tr>
<td>w/o sparse features</td>
<td>28.2 (+.3)</td>
<td>20.4 (+.1)</td>
<td>29.6 (+.1)</td>
<td>21.7 (+.1)</td>
<td>32.2 (+.1)</td>
<td>30.0 (+.3)</td>
<td></td>
<td>-2</td>
</tr>
<tr>
<td>w/o language model</td>
<td>28.3 (+.2)</td>
<td>20.5 (+.0)</td>
<td>29.5 (+.2)</td>
<td>21.4 (+.4)</td>
<td>31.5 (+.8)</td>
<td>29.2 (+.6)</td>
<td></td>
<td>-4</td>
</tr>
<tr>
<td>w/o reordering model</td>
<td>28.5 (+.0)</td>
<td>20.5 (+.0)</td>
<td>-</td>
<td>21.8 (+.0)</td>
<td>32.3 (+.0)</td>
<td>29.8 (+.1)</td>
<td></td>
<td>±0</td>
</tr>
<tr>
<td>w/o operation sequence model</td>
<td>28.3 (+.2)</td>
<td>20.3 (+.1)</td>
<td>29.7 (+.0)</td>
<td>21.7 (+.1)</td>
<td>32.0 (+.3)</td>
<td>29.5 (+.2)</td>
<td></td>
<td>-2</td>
</tr>
</tbody>
</table>

Model are a 5-gram LM score, phrase translation and lexical translation scores, word and phrase penalties, and a linear distortion score. The phrase translation probabilities are smoothed with Good-Turing smoothing (Foster et al., 2006). We used the hierarchical lexicalized reordering model (Galley and Manning, 2008) with 4 possible orientations (monotone, swap, discontinuous left and discontinuous right) in both left-to-right and right-to-left direction. We also used the operation sequence model (OSM) (Durrani et al., 2013) with 4 count based supportive features. We further employed domain indicator features (marking which training corpus each phrase pair was found in), binary phrase count indicator features, sparse phrase length features, and sparse source word deletion, target word insertion, and word translation features (limited to the top $K$ words in each language, typically with $K = 50$).

### 3.5 Tuning

Since our feature set (generally around 500 to 1000 features) was too large for MERT, we used $k$-best batch MIRA for tuning (Cherry and Foster, 2012). To speed up tuning we applied threshold pruning to the phrase table, based on the direct translation model probability.

### 3.6 Decoding

In decoding we applied cube pruning (Huang and Chiang, 2007) with a stack size of 5000 (reduced to 1000 for tuning), Minimum Bayes Risk decoding (Kumar and Byrne, 2004), a maximum phrase length of 5, a distortion limit of 6, 100-best translation options and the no-reordering-over-punctuation heuristic (Koehn and Haddow, 2009).

### 4 Experimental Results

In this section we describe peculiarities of individual systems and present experimental results.

#### 4.1 French↔English

Our submitted systems for the French-English language pair are quite similar for the two translation directions. We used all the constrained parallel data to build a phrase-based translation model and the language model was build from the target side of this data, the monolingual news data and the LDC GigaWord corpora. During system development we used the newdiscussdev2015 for tuning and development testing, using 2-fold cross validation. For tuning the submitted system, and the post-submission experiments, we tuned on the whole of newdiscussdev2015, and report cased BLEU on newdiscusstest2015.

Prior to submission we experimented with bilingual LM and an NPLM-based neural network language model (Sections 2.2 and 2.1) but did not obtain positive results. These were trained on a randomly selected 10% portion of the parallel training data. We also experimented with class-based language models (using Och clusters from mkcls), including the 50 class language model in the English→French submission but not in the French→English one, since it helped in our development setup in the former but not the latter.

In the post-submission experiments (Table 2), we show the comparison of the baseline system (as described in Section 3) with systems enhanced with bilingual LM, NPLM and class-based language models. For the class-based language models, we tested with 50 Och clusters, 200 Och clusters, and with both class-based LMs. For the bilingual LM, we created both “combined” (a 5-gram on the target and a 9-gram on the source) and “source” (1-gram on the target and 15-gram on the source).
source) models. The bilingual LMs are trained on 10% of the available parallel data, selected using modified Moore-Lewis data selection (Moore and Lewis, 2010; Axelrod et al., 2011). The NPLM is a 5-gram model trained on all available language model data.

We observe from Table 2 that the bilingual LM has a minimal effect on BLEU, only showing an increase for one language pair, one configuration, and the margin of improvement is probably within the margin of tuning variation. We do not have a good explanation for the lack of success with bilingual LM, in contrast to (Devlin et al., 2014), however we note that all reports of improvements with this type of model are for distantly related language pairs. We also did not observe any improvement with the class-based language models for French→English, although we did observe small gains from English→French. Building an NPLM model for all data gives a reasonable improvement (+0.7) for the French target, but not the English. In fact French→English was the only language pair where NPLM did not improve BLEU after building the LM on all data. It is possible that the limited morphology of English means that the improved generalisation of the NPLM is not as helpful, and also that the conventional n-gram LM is already strong for this language pair.

### 4.2 Finnish→English

For the Finnish-English language pair we built systems using only the constrained data, and systems using all the OPUS (Tiedemann, 2012) parallel data. Our baselines include this extra data, but we also show results just using the constrained parallel data. We did not employ the morphological splitting as in Edinburgh’s syntax-based system (Williams et al., 2015) and consequently the English→Finnish systems performed poorly in development and we did not submit a phrase-based system for this pair.

Our development setup was similar to French→English; we used the newsdev2015 for tuning and test during system development (in 2-fold cross-validation) then for the submission and subsequent experiments we used the whole of newsdev2015 for tuning. Also in common with our work on French→English, we performed several post-submission experiments to examine the effect of class-based language models, bilingual LM and NPLM. We show the results in Table 3. For training bilingual LM and NPLM models we encountered some numerical issues, probably due to the large vocabulary size in Finnish. These were partially addressed by employing dropout to prevent overfitting (Srivastava et al., 2014), enabling us to train the models for at least 2 epochs.

We note that, as with French→English, our application of bilingual LM did not result in significant improvement. Finnish and English are quite distantly related, but we can speculate that using words as a representation for Finnish is not appropriate. The NPLM, however, offers modest (+0.4) improvements over the baseline in both directions.
4.3 Czech ↔ English

The development of the Czech ↔ English systems followed the ideas in Section 2.3, i.e., with a focus on word classes (50, 200, 600 classes) for all component models. We combined the test sets from 2008 to 2012 for tuning. No neural language model or bilingual language model was used.

4.4 Russian ↔ English

To Russian. For the English ↔ Russian system, we used all the parallel data specified in the task. The Wiki Headlines data was appended onto the combined parallel corpus. For the monolingual corpora, we used all the constrained track corpora except for News crawl 2008–2010 which were overlooked as they were much smaller than other resources. We trained word classes with three different settings (50, 200, and 600 clusters) on both source and target languages. On applying clusters, we trained 6-gram language models on the target side. We used all four factors (words and clusters) in both source and target languages for the translation model and the OSM, but we used only the word factor for the alignment and the reordering models. We performed transliteration (Durrani et al., 2014c) after decoding for all three experimental conditions. We used news test 2012 for LM interpolation and batch MIRA model tuning. In Table 4, the only difference between the baseline system and the official submission is that the baseline has no cluster factors. The final model (BiLM source & combined & NPLM) is the same as the submitted system, apart from the fact that we applied two bilingual neural network models: one over the source and one over the source and target, and an NPLM language model over the target. This did not improve over the factored model and so was not submitted for the evaluation.

From Russian. The Russian ↔ English system used the same settings as the Czech system, except for the addition of a factor over 2000 word classes and a smaller tuning set (just news test 2012).

4.5 German ↔ English

Our German–English training corpus comprises all permissible parallel data of the constrained track for this language pair. A concatenation of newsyscomb2009 and news test 2008–2012 served as tuning set.

<table>
<thead>
<tr>
<th>System</th>
<th>en-ru</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>25.0</td>
</tr>
<tr>
<td>Submitted</td>
<td>25.2</td>
</tr>
<tr>
<td>BiLM source &amp; combined &amp; NPLM</td>
<td>25.1</td>
</tr>
</tbody>
</table>

Table 4: Experimental results (cased BLEU) for English—Russian averaged over news test 2013 and news test 2014.

From German. For translation from German, we applied syntactic pre-reordering (Collins et al., 2005) and compound splitting (Koehn and Knight, 2003) in a preprocessing step on the source side. A rich set of translation factors was exploited in addition to word surface forms: Och clusters (50 classes), morphological tags, part-of-speech tags, and word stems on the German side (Schmid, 2000), as well as Och clusters (50 classes), part-of-speech tags (Ratnaparkhi, 1996), and word stems (Porter, 1980) on the English side. The factors were utilized in the translation model and in OSMs. The lexicalized reordering model was trained on stems. Individual 7-gram Och cluster LMs were trained with KenLM’s --discount_fallback --prune ‘0 0 1’ parameters, then interpolated with the SRILM toolkit and added to the log-linear model as a second LM feature. Our 5-gram word LM was trained on all English data at once, also with pruning of singleton n-grams of order 3 and higher. We included the English LDC Gigaword Fifth Edition. Sparse lexical features (source word deletion, target word insertion, word translation) were limited to the top $K = 200$ words for German—English.

To German. Translation factors for the English—German translation direction are word surface forms, Och clusters (50 classes), morphological tags, and part-of-speech tags. Morphological tags were employed on the target side only, all other factors on both source and target side. The lexicalized reordering model was trained on word surface forms. We added an interpolated 7-gram Och cluster LM and a 7-gram LM over morphological tags. LMs were trained in a similar way as the ones for translation from German. Sparse phrase length features and sparse lexical features were not used for English—German.

---

Sparse lexicalized reordering. We investigated sparse lexicalized reordering features (Section 2.4) on the German-English language pair in both translation directions. Two methods for learning the weights of the sparse lexicalized reordering feature set have been compared: (1) direct tuning in MIRA along with all other features in the model combination (sparse LR (MIRA)), and (2) separate optimization with stochastic gradient descent (SGD) with a maximum expected BLEU objective (sparse LR (SGD)). For the latter variant, we used the MT tuning set for training (13,573 sentence pairs) and otherwise followed the approach outlined by Auli et al. (2014). We tuned the baseline feature weights with MIRA before SGD training and ran two final MIRA iterations after it. SGD training was stopped after 80 epochs.

Empirical results for the German-English language pair are presented in Table 5. We observe minor gains of up to +0.2 points BLEU. The results are not consistent in the two translation directions: The MIRA-trained variant seems to perform better when translating from German, the SGD-trained variant when translating to German. However, in both cases the baseline score is almost identical to the best results with sparse lexicalized reordering features.

In future work we plan to adopt hypergraph MIRA, as well as larger training sets for maximum expected BLEU training. We also consider scaling the method to word surface forms in addition to Och clusters, and trying RPROP instead of SGD.

5 Conclusion


Acknowledgements

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreements 645452 (QT21), 645487 (MMT), 644333 (TraMOOC) and 644402 (HimL).

References


Meeting of the Association for Computational Linguistics (ACL ’05), pages 531–540, Ann Arbor, MI, USA, June.


