Abstract

This paper describes the University of Edinburgh’s machine translation (MT) systems for the IWSLT 2015 evaluation campaign. Our submissions are based on preliminary systems which are under development for the purpose of lecture translation in the TraMOOC project, funded by the European Union. We participated in the English→Chinese and the English→German translation tasks in the MT track, utilizing only data supplied by the organizers or listed as permissible. We built phrase-based translation systems for both tasks. For English→German, we furthermore made use of syntax-based translation and system combination.

1. Introduction

The University of Edinburgh’s translation engines are based on the open source Moses toolkit [1]. We set up phrase-based systems [2, 3] for the English→Chinese and English→German translation tasks, and additionally a string-to-tree syntax-based system [4, 5] for English→German. Our primary submission translations for English→Chinese are the output of a single phrase-based system, whereas our primary submission translations for English→German are the output of a system combination [6] of two phrase-based systems and one syntax-based system.

The setups for our phrase-based systems have evolved from the configurations of the engines we built for Edinburgh’s participation in last year’s IWSLT evaluation [7] and in this year’s Workshop on Statistical Machine Translation (WMT) shared translation task [8].

Edinburgh’s syntax-based systems have recently yielded state-of-the-art performance on English→German news translation tasks [9, 10] and have been applied in an IWSLT-style setting for the first time for our last year’s contrastive submission [7]. This year, a syntax-based system became part of our primary submission by contributing input to a system combination.

For system combination, we employed the implementation that has been released as part of the Jane machine translation toolkit [11]. Multiple previous top-ranked submissions to open evaluation campaigns have relied on this system combination framework [12, 13, 14].

2. System Overview

2.1. Training and Tuning

For both the phrase-based systems and the syntax-based system, we first preprocess the parallel training data and then create word alignments by aligning the data in both directions with MGIZA++ [15]. We use a sequence of IBM word alignment models [16] with five iterations of EM training [17] of Model 1, three iterations of Model 3, and three iterations of Model 4. After EM, we obtain a symmetrized alignment by applying the grow-diag-final-and heuristic [18, 3] to the two trained alignments. We extract bilingual phrases that are consistent with the symmetrized word alignment from the parallel training data. In the case of the syntax-based system, we also need syntactic parses of the target-language side of the parallel training data in order to extract synchronous context-free grammar rules.

We train n-gram language models (LMs) with modified Kneser-Ney smoothing [19, 20]. KenLM [21] is employed for LM training and scoring, and SRILM [22] for linear LM interpolation.

Our translation model incorporates a number of different features in a log-linear combination [23]. We tune the feature weights with batch k-best MIRA [24] to maximize BLEU [25] on a development set. We run MIRA for 25 iterations on 200-best lists (phrase-based) or 1000-best lists (syntax-based).

In our experiments (cf. Section 3) with the phrase-based system, we commence with a plain baseline which comprises a small amount of vital features only. We then incrementally extend the system with further features and more advanced techniques. Each setup is re-tuned individually to obtain optimal feature weights for the respective configuration.
2.2. Phrase-based System

The features of our plain phrase-based baseline are:

- Phrase translation log-probabilities in both target-to-source and source-to-target direction.
- Lexical translation log-probabilities in both target-to-source and source-to-target direction.
- Word penalty.
- Phrase penalty.
- A distance-based distortion cost.
- A 5-gram language model over words. Singleton n-grams of order three and higher are discarded.

We extract phrases up to a length of five. We prune the phrase table to a maximum of 100 best translation options per distinct source side and apply a minimum score threshold \( \tau \) on the source-to-target phrase translation probability, with \( \tau = 0.0001 \) during tuning and \( \tau = 0.00001 \) during testing. We use cube pruning [26] in decoding. Pop limit and stack limit are set to 1000 for tuning and to 5000 for testing. A distortion limit of six is enforced during decoding, and we disable reordering over punctuation. Furthermore, Minimum Bayes Risk decoding [27] is employed for testing. Extensions we experimented with for either English→German or English→Chinese are:

**LRM.** A hierarchical lexicalized reordering model [28]. This model estimates the probabilities of orientation classes for each phrase from the training data. We use four orientation classes: monotone, swap, left-discontinuous, and right-discontinuous.

**TM factors.** Translation model (TM) factors beyond word surface forms [29, 30]. Factors can for instance be part-of-speech (POS) tag, morphological tag, or automatically learnt word classes, e.g. from *mkcls* [31]. Factors can be added on either source side or target side or both. We do not use a generation step but merely enrich the phrases with factored annotation. The annotation is obtained by tagging the training data prior to phrase extraction. Source-side factors such as POS or morphological tags can be helpful for disambiguating phrases: at decoding time, we annotate the input text in a preprocessing step, and the decoder only applies phrases with matching annotation. Target-side factors can be helpful for providing a longer context window via n-gram models of higher order over representations given by the factors (which we mention next in this list).

**7-gram class-based L.M.** A 7-gram language model over *mkcls* word classes.

**7-gram POS L.M.** A 7-gram language model over part-of-speech tags.

**7-gram morph L.M.** A 7-gram language model over morphological tags.

**Good-Turing smoothing.** Good-Turing smoothing of phrase translation probabilities [32].

**Count features.** Seven binary features indicating absolute occurrence count classes of phrase pairs.

**Sparse features.** Sparse phrase length features, and sparse lexical features for the top 200 words.

**Domain indicators.** Binary features indicating the provenance of phrase pairs: if a phrase pair has been seen in a particular training corpus, a binary indicator associated with the respective training corpus fires on application of that phrase pair during decoding.

**Phrase table fill-up.** A foreground phrase table extracted from in-domain data is filled up with entries from a background phrase table extracted from all data [33, 34]. An entry from the background table is only added if the foreground table does not know the respective phrase identity. A binary feature distinguishes background phrases from foreground phrases. (The baseline uses a phrase table extracted from all data.)

**5-gram OSM.** A 5-gram operation sequence model [35].

**5-gram OSM over word classes.** A 5-gram operation sequence model over *mkcls* word classes.

**5-gram OSMs over factors.** Operation sequence models over various representations given by the factors.

**In-domain OSMs.** 5-gram operation sequence models over words and factors, trained on the in-domain portion of the parallel data only.

**Unpruned L.M.** The baseline 5-gram language model over words is replaced by a version where singleton n-grams of order three and higher have not been discarded.

**No singleton phrases.** Phrase pairs with an occurrence count of one are removed from the phrase table.

**Sparse L.R.** Sparse lexicalized reordering features [36] with weights learnt via RPROP with a maximum expected BLEU objective [37, 38]. The features are added on top of the standard hierarchical lexicalized reordering model. We apply features based on all words as well as word classes with 200 clusters on both source and target side. Active feature groups are between, phrase, and stack. We follow a similar training procedure as suggested by Wuebker et al. [38]. Maximum expected BLEU training with RPROP is conducted on the in-domain fraction of the training data. We train on 100-best lists. We set the regularization parameter to \( 10^{-3} \) and use the weights obtained after 50 iterations of RPROP. Rather than decoding the training data with leaving-one-out, we utilize a system with no singleton phrases. The learnt sparse lexicalized reordering features are condensed to a single feature per orientation, as suggested by Auli et al. [37]. A final MIRA run tunes weights for those condensed features along with the other features in the log-linear model of the translation system.

---

Our tool for maximum expected BLEU training has been released as part of the Moses code base on GitHub.
2.3. Syntax-based System

The syntactic translation model for our string-to-tree system conforms to the GHKM syntax approach as proposed by Galley, Hopkins, Knight, and Marcu [4] with composed rules [39, 40]. Decoding is carried out with a procedure based on bottom-up chart parsing. The parsing algorithm is extended to handle translation candidates and to incorporate language model scores via cube pruning [26].

Standard features of Edinburgh’s string-to-tree syntax-based systems are:

- Rule translation log-probabilities in both target-to-source and source-to-target direction, smoothed with Good-Turing discounting.
- Lexical translation log-probabilities in both target-to-source and source-to-target direction.
- Word penalty.
- Rule penalty.
- A rule rareness penalty.
- The monolingual PCFG probability of the tree fragment from which the rule was extracted.
- A 5-gram language model over words.

When extracting syntactic rules, we impose several restrictions for composed rules, in particular a maximum number of 100 tree nodes per rule, a maximum depth of seven, and a maximum size of seven. We discard rules with non-terminals on their right-hand side if they are singletons in the training data. Only the 200 best translation options per distinct rule source side with respect to the weighted rule-level model scores are loaded by the decoder. Search is carried out with a maximum chart span of 25, a rule limit of 500, a stack limit of 200, and a pop limit of 1000 for cube pruning [41]. During tuning, we constrain the translation options per rule source side to the top 20 candidates for faster optimization, and we set the cube pruning pop limit to 500. We configure Moses’ n-best-factor parameter at a value of 100 to avoid short n-best lists.

For our IWSLT English→German syntax-based system, the target side of the parallel training data is parsed with BitPar [42]. We remove grammatical case and function information from the annotation obtained with BitPar and apply right binarization of the German parse trees prior to rule extraction [43, 44, 45].

The system is adapted to the TED domain by extracting two separate rule tables (from in-domain data and from out-of-domain parallel data) and merging them with a fill-up technique [33]. We also integrate a second 5-gram LM trained on the in-domain corpus into the log-linear combination. Additionally we add soft source syntactic constraints [46] and augment the system with non-syntactic phrases [47].

2.4. System Combination

The Jane machine translation toolkit implements a system combination approach via confusion network decoding [11]. The hypotheses from individual MT systems are aligned to each other with METEOR [48]. A confusion network is generated which represents all combined translations that can be produced from the set of individual hypotheses. The optimal combined hypothesis is chosen by finding the best path through the confusion network. The decision process is guided by a couple of simple features:

- Binary system voting features.
- A primary system indicator.
- Word penalty.
- A small 3-gram language model trained only on the set of individual hypotheses.
- A conventional 5-gram language model.

Feature weights are optimized with MERT [49].

We combine three individual systems with this method for our English→German primary submission.

3. Experiments

3.1. English→German MT

For the English→German MT task, we submitted outputs of two different phrase-based systems (contrastive 1 and contrastive 2), a syntax-based system (contrastive 3), and a system combination (primary) of those three single systems. Table 1 shows their respective performance in terms of BLEU scores, along with the official scores [50] of the best last year’s submission for comparison.

Our English→German systems are trained using monolingual and parallel data from the in-domain WIT corpus [52], as well as Europarl [53], MultiUN [54], the parallel corpus from the Wikipedia [55] as provided for the evaluation campaign, the German Political Speeches corpus [56], and the permissible corpora from the WMT shared translation task [57]. For the systems with factors, annotation exploited in addition to word surface forms is: part-of-speech tags [58] on the English side; morphological tags [59] and part-of-speech tags [59] on the German side; and word classes from mkcls with 50 clusters on both sides.

5-gram LMs over words are estimated over a concatenation of all target-language training data, rather than linearly interpolating individual LMs over the different corpora. We found this to perform equally well or better on the given task. Class-based LMs, POS LMs, and morph LMs, on the other hand, are linear interpolations of individually trained LMs.3 Feature weights for all single engines are tuned on a concatenation of TED.dev2010, TED.tst2010, TEDX.dev2012, and TEDX.tst2013.4

3Individual LMs over factors are trained with KenLM’s --discount_fallback --prune 0 0 1 parameters.
4Note that TEDX.tst2013 and tst2013 (=TED.tst2013) are two different sets.
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>phrase-based</td>
<td>28.3</td>
<td>24.7</td>
<td>26.3</td>
<td>23.3</td>
<td>25.4</td>
</tr>
<tr>
<td>(+) w/o singleton</td>
<td>27.9</td>
<td>24.5</td>
<td>26.8</td>
<td>23.3</td>
<td>25.5</td>
</tr>
<tr>
<td>phrases + sparse LR</td>
<td>26.8</td>
<td>23.6</td>
<td>26.1</td>
<td>22.7</td>
<td>24.3</td>
</tr>
<tr>
<td>system combination</td>
<td>28.4</td>
<td>25.6</td>
<td>27.0</td>
<td>24.0</td>
<td>26.0</td>
</tr>
<tr>
<td>best IWSLT</td>
<td>–</td>
<td>–</td>
<td>26.2</td>
<td>23.3</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 1: Edinburgh submission system results for the English→German MT task (case-sensitive BLEU scores), and results of the best IWSLT 2014 submission as reported by Cettolo et al. [50]. The Edinburgh primary submission is a system combination of the three contrastive systems and was tuned on tst2012.

<table>
<thead>
<tr>
<th>en→zh</th>
<th>tst2012</th>
<th>tst2013</th>
<th>tst2014</th>
<th>tst2015</th>
</tr>
</thead>
<tbody>
<tr>
<td>phrase-based</td>
<td>21.3</td>
<td>22.9</td>
<td>19.6</td>
<td>25.4</td>
</tr>
<tr>
<td>system combination</td>
<td>–</td>
<td>22.5</td>
<td>21.6</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 2: Edinburgh submission system results for the English→Chinese MT task (character-based BLEU scores), and results of the best IWSLT 2014 submission as reported by Cettolo et al. [50].

**Phrase-based system.** Table 3 presents the results achieved with the plain phrase-based baseline, and the gains when incrementally adding extensions as described in Section 2.2.5 The contrastive 1 submission system outperforms the plain baseline by up to +3.6 BLEU points (on tst2011). If we remove singleton phrases on top of that, we observe a small gain on tst2013, but performance degrades slightly on tst2011 and tst2012. The sparse lexicalized reordering features trained via RPROP with a maximum expected BLEU objective (contrastive 2) do not further affect the results too much.6 However, the contrastive 2 submission system outperforms the plain baseline by +3.5 BLEU points on a different test set (on tst2013).

**Syntax-based system.** In the syntax-based system, we utilize neither the parallel corpus from the Wikipedia nor MultiUN or the German Political Speeches corpus for rule extraction. We only use the target side of the Wikipedia corpus as LM training data. The development set is the same as for the phrase-based systems. Our IWSLT string-to-tree syntax-based system (contrastive 3) is outperformed by the phrase-based submission systems by a bit more than one BLEU point on this year’s evaluation set (tst2015), cf. Table 1. The average BLEU delta on the other test sets is lower, though.

**System combination.** The parameters of the system combination (primary) are optimized on tst2012. The consensus translation produced by the system combination boosts the BLEU score by half a point over the best single system on this year’s evaluation set (tst2015), cf. Table 1. Improvements on the other test sets vary between +0.1 and +0.5 BLEU points when incrementally adding extensions as described in Section 2.2.5

<table>
<thead>
<tr>
<th>en→de</th>
<th>tst2011</th>
<th>tst2012</th>
<th>tst2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>phrase-based baseline</td>
<td>24.7</td>
<td>22.0</td>
<td>23.3</td>
</tr>
<tr>
<td>+ LRM</td>
<td>25.5</td>
<td>22.0</td>
<td>24.1</td>
</tr>
<tr>
<td>+ TM factors</td>
<td>25.3</td>
<td>22.1</td>
<td>23.8</td>
</tr>
<tr>
<td>+ 7-gram class-based LM</td>
<td>25.9</td>
<td>22.5</td>
<td>24.2</td>
</tr>
<tr>
<td>+ 7-gram POS LM</td>
<td>26.1</td>
<td>22.8</td>
<td>24.6</td>
</tr>
<tr>
<td>+ 7-gram morph LM</td>
<td>26.5</td>
<td>22.9</td>
<td>24.9</td>
</tr>
<tr>
<td>+ Good-Turing smoothing</td>
<td>26.8</td>
<td>23.6</td>
<td>24.9</td>
</tr>
<tr>
<td>+ count features</td>
<td>26.8</td>
<td>23.4</td>
<td>24.9</td>
</tr>
<tr>
<td>+ sparse features</td>
<td>26.9</td>
<td>23.7</td>
<td>25.1</td>
</tr>
<tr>
<td>+ domain indicators</td>
<td>27.2</td>
<td>23.6</td>
<td>25.3</td>
</tr>
<tr>
<td>+ 5-gram OSM</td>
<td>27.6</td>
<td>24.1</td>
<td>26.1</td>
</tr>
<tr>
<td>+ 5-gram OSMs over factors</td>
<td>27.8</td>
<td>24.3</td>
<td>26.0</td>
</tr>
<tr>
<td>+ in-domain OSMs</td>
<td>28.0</td>
<td>24.3</td>
<td>26.3</td>
</tr>
<tr>
<td>+ unpruned LM (+)</td>
<td>28.3</td>
<td>24.7</td>
<td>26.3</td>
</tr>
<tr>
<td>+ no singleton phrases</td>
<td>27.9</td>
<td>24.6</td>
<td>26.7</td>
</tr>
<tr>
<td>+ sparse LR (+)</td>
<td>27.9</td>
<td>24.5</td>
<td>26.8</td>
</tr>
</tbody>
</table>

Table 3: Incremental improvements over a plain phrase-based baseline for English→German (case-sensitive BLEU scores).

<table>
<thead>
<tr>
<th>en→zh</th>
<th>tst2012</th>
<th>tst2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>phrase-based baseline</td>
<td>19.2</td>
<td>21.0</td>
</tr>
<tr>
<td>+ LRM</td>
<td>19.8</td>
<td>21.7</td>
</tr>
<tr>
<td>+ Good-Turing smoothing</td>
<td>20.0</td>
<td>21.9</td>
</tr>
<tr>
<td>+ count features</td>
<td>20.1</td>
<td>21.9</td>
</tr>
<tr>
<td>+ 7-gram class-based LM (in-domain)</td>
<td>20.0</td>
<td>22.0</td>
</tr>
<tr>
<td>+ phrase table fill-up</td>
<td>21.0</td>
<td>22.3</td>
</tr>
<tr>
<td>+ 5-gram OSM</td>
<td>21.0</td>
<td>22.5</td>
</tr>
<tr>
<td>+ 5-gram OSM over word classes</td>
<td>20.9</td>
<td>22.5</td>
</tr>
<tr>
<td>+ in-domain OSMs (primary)</td>
<td>21.3</td>
<td>22.9</td>
</tr>
</tbody>
</table>

Table 4: Incremental improvements over a plain phrase-based baseline for English→Chinese (character-based BLEU scores).
Our best single system yields translation quality on the level of the last year’s best submission, which was a system combination [14]. Our primary submission is around 0.7 BLEU points better than last year’s best submission.

3.2. English→Chinese MT

For the English→Chinese MT task, we submitted the output of a phrase-based single system (primary). Table 2 shows the performance in terms of BLEU scores, measured on character level with the aid of the Chinese character tokenization script provided by the organizers of the evaluation campaign. For comparison, we also include the official scores [50] of the best last year’s submission.

Our English→Chinese systems are trained using monolingual and parallel data from the in-domain WIT3 corpus [52], as well as MultiUN [54]. For the English-Chinese MultiUN parallel data, we resorted to the sentence-aligned version as distributed in OPUS [60]. We perform Chinese word segmentation with the Stanford Word Segmenter [61] as a preprocessing step on all target-side data. The character-based tokenization is conducted for evaluation purposes only, whereas our models operate on word-segmented data.

Table 4 presents the results achieved with the plain phrase-based baseline, and the gains when incrementally adding extensions as described in Section 2.2. The 5-gram LM over words is a linear interpolation of individual LMs, the 7-gram class-based LM is trained on in-domain data only. The only factors we use for English→Chinese are word classes from mkcls with 50 clusters. Feature weights are tuned on a concatenation of dev2010, tst2010, and tst2011. The submission system outperforms the plain baseline by up to +2.1 BLEU points (on tst2012).

The comparison with last year’s best submission [51] is somewhat surprising: the BLEU score of our system is +0.4 points higher on tst2013, but we significantly lag behind on tst2014. We are currently unaware of the reason for this behavior.

4. Summary

We built high-quality machine translation systems for the IWSLT 2015 English→Chinese and English→German translation tasks in the MT track. By utilizing advanced features and techniques, we have been able to achieve improvements over plain phrase-based baselines of two BLEU points or more on both language pairs. All methods we employed are implemented in publicly available software such as the Moses and the Jane statistical machine translation toolkits.

5. Acknowledgements

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement № 644333 (TraMOOC).

6. References


Proceedings of the 12th International Workshop on Spoken Language Translation
Da Nang, Vietnam, December 3–4, 2015


M. Denkowski and A. Lavie, “Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of


