Sheffield Systems for the English-Romanian Translation Task

Frédéric Blain **Xingyi Song** **Lucia Specia**
Department of Computer Science
The University of Sheffield
United Kingdom
{f.blain,xsong2,l.specia}@sheffield.ac.uk

Abstract

This paper provides an overview of the submissions the University of Sheffield for the English-Romanian Translation Task of the ACL 2016 First Conference on Machine Translation (WMT16). The submitted translations were produced with a phrase-based system trained using the Moses toolkit, in two variants: (i) n-best rescoring using additional features from Quality Estimation (primary submission), and (ii) a novel weighted ranking optimisation approach (secondary submission).

1 Introduction

This paper presents the submissions the University of Sheffield for the shared translation task, which is part of the ACL 2016 First Conference on Machine Translation (WMT16). We participated in the English-Romanian language pair.

Our primary submission investigates the use of additional features from Quality Estimation (QE) to better discriminate translation hypothesis within an n-best list produced by a phrase-based MT system built with the Moses toolkit (Koehn et al., 2007). The idea is to expand the n-best list feature set with additional features coming not from the MT system, but from external resources. Our expectation is that external, potentially richer features could help guide the decoder to produce better quality translations.

In addition to our primary system, we investigate the use of a different optimisation algorithm to tune the parameters of our phrase-based SMT system: the Weighted Ranking Optimisation (WRO) algorithm. Derived from the Pairwise Ranking Optimisation (PRO) algorithm (Hopkins and May, 2011), WRO addresses various limitations of PRO, as we discuss in Section 4.

In the following section we describe the settings of our phrase-based MT system. The two versions of our phrase-based system are presented in Section 3 and 4, respectively. We report our results on the newstest2016 test set in Section 5.

2 USFD Phrase-based System

We only used the data that was made officially available for the English-Romanian task (constrained submission). Statistics of the available training resources for the task are given in table 1.

As pre-processing, the English part of the data was tokenised using the Moses tokenisation script, while the Romanian part was tokenised using Tokro[1] (Allauzen et al., 2016), a rule-based tokeniser that normalises diacritics and splits punctuation and clitics.

Our phrase-based model was trained following the standard "baseline" settings of the Moses toolkit with MGIZA (Gao and Vogel, 2008) for word alignment and KenLM (Heafield, 2011) for language modelling. The phrase length was limited to 5. Lexicalised reordering models were trained using the same data.

We built a 5-gram Romanian language model (LM) from the linear interpolation of four individual LMs. The two first were built on the target side of the in-domain parallel corpora (Europarl7, SETimes2). For the two last, we use subsets of both the News Commentary (93%) and the Common Crawl (13%), selected using XenC-v2.1[2] (Rousseau, 2013) in mode 2[3] with the parallel corpora (Europarl7, SETimes2) as in-domain data.

[1] https://perso.limsi.fr/aufrant/software/tokro

[2] https://github.com/rousseau-lium/XenC/

[3] Implementation of the Moore-Lewis cross-entropy filtering method

Proceedings of the First Conference on Machine Translation, Volume 2: Shared Task Papers, pages 259–263,
Berlin, Germany, August 11-12, 2016.

	English		Romanian	
	# seg	# word	# seg	# word
Parallel data				
Europarl7	394k	10.4M	394k	10.4M
SETimes2	211k	5.03M	211k	5.36M
Monolingual data for language modelling				
News Commentary			2.28M	55.1M
– selected with XenC: 93%			2.1M	52.2M
Common Crawl			289M	7.93G
– selected with XenC: 13%			23.7M	577M
Development data				
newsdev_1	1k	24.7k	1k	26.7k
newsdev_2	1k	25.2k	1k	25.6k
setimes2	2k	47.8k	2k	50.9k

Table 1: Statistics of the available data for the English-Romanian Machine Translation Task (constrained submission). For our language modelling we only used 93% and 13% of the News Commentary and the Common Crawl corpus, respectively, after data selection.

The optimisation of the parameters was achieved using a 100-best Minimum Error Rate Training (MERT) (Och, 2003) towards the BLEU metric (Papineni et al., 2002).

3 N-best Rescoring with QE Features

Quality Estimation (QE) aims at measuring the quality of the Machine Translation (MT) output without reference translations. Generally, QE is addressed with various features indicating fluency, adequacy and complexity of the source-translation text pair. We hypothesise that these could help discriminate translation hypothesis in an n-best list.

In our scenario, we first generate 1000 distinct n-best translation candidates using the phrase-based system described in Section 2. For each translation candidate, we extend its feature set by adding 17 new features corresponding to the baseline `black-box` QE features[4] extracted with the QuEst++ toolkit[5] (Specia et al., 2015).

The baseline `black-box` feature extraction process does not require to train a complete QE system. For that, QuEst++ only requires some resources: both source and target language models, source-target lexical table, and n-gram counts. In our case we use the same data as for our phrase-base SMT system in order to generate these resources.

Given the updated n-best with the decoder and QE features, we use the rescoring scripts available within the Moses toolkit[6] to learn new feature weights on a development set using the k-best MIRA algorithm (Cherry and Foster, 2012). Finally, the 1000-best list with distinct translations generated from the test set are rescored, re-ranked and the 1-best translation is used as final translation candidate.

4 Weighted Ranking Optimisation

The Weighted Ranking Optimisation algorithm is based on PRO. PRO estimates weights by classifying translation candidate pairs in the n-best list into "correctly ordered" and "incorrectly ordered" according an automatic evaluation metric. However, enumerating all possible pairs in the n-best is impractical, even with a small 100-best list the number of pairs still makes it impractical. PRO uses a sampling strategy to avoid this problem. The sampling strategy first randomly selects a Γ number of candidate pairs, and further select Ξ pairs of candidate with the largest metric difference. The model weights are then trained using the MegaM (Daume, 2004) classifier with the selected samples.

WRO uses same procedure as PRO, but with a different sampling strategy. Also, it uses a different weighting scheme for the training samples. In a nutshell, WRO aims to address the following limitations of PRO:

1. PRO's random sampling is not the optimum way for selecting samples since the target is not clear. As we only select a small sample from the entire sample space, a clearer target should give better training quality. We refer to these targets as `oracles`, as they are the translation output we want the system to produce, often the reference translation. In WRO, the oracles are the top 10% candidates (sorted by BLEU score) in the n-best list;

2. In PRO all sampled sentences are considered equally important. Although we select the same number of samples for each training

[4] `www.quest.dcs.shef.ac.uk/quest_files/features_blackbox_baseline_17`

[5] `www.quest.dcs.shef.ac.uk`

[6] `http://github.com/moses-smt/mosesdecoder/tree/master/scripts/nbest-rescore`

sentence, these sentences may be very different. For example, reachable sentences[7] can be more important than unreachable ones. Unreachable translations are very common in SMT. They may be caused by words in the reference translation that do not appear in the system's phrase table, i.e. that have not been seen (enough) in the training corpus. This could also happen because the reference translation is inherently wrong, which is common in crowd-sourced corpora. In both cases, unreachable translations cannot be correctly scored by automatic evaluation metrics. Therefore, we cannot learn useful information from unreachable translations to discriminate between good and bad translations, and this often harms training performance.

3. PRO uses BLEU to assess the quality of translation candidates. However, BLEU was originally designed for document-level evaluation, and as such is less accurate for sentence-level evaluation.

The WRO procedure is described in Algorithm 1 with SIMPBLEU_RECALL (Song et al., 2013) used as the scoring function for the evaluation of translation candidates. In previous WMT editions (Callison-Burch et al., 2012; Macháček and Bojar, 2013), SIMPBLEU has been shown to have better correlation than BLEU for the assessment of translation quality at sentence-level.

Similar to PRO, we use n-best list Nb as one of our candidate pools for sample selection. We also create another list called oracle list, Nb_{oracle}. We select the top 10 percent of all candidates in the n-best list with the highest metric score as oracles and store them in the oracle list.

The sampling procedure includes two steps: first, a Γ number of candidate pairs $\{e_s, e'_s\}$ are randomly selected from the two lists, where e_s and e'_s are represented by their corresponding feature values $h(e_s)$ and $h(e'_s)$. Contrary to PRO, WRO focuses on ranking the oracle translations in the correct order among all candidates. In this case, we define the candidate e_s as an oracle that is randomly selected from the oracle list Nb_{oracle}, and e'_s is the non-oracle that is randomly selected from the n-best list Nb. We select e'_s from entire n-best list (if e'_s is also included in the top 10% candidates with highest metric score, then the candidate with the better metric score is considered the oracle). The selected candidates are then evaluated by an automatic evaluation metric m. Sampled pairs with a metric difference (i.e. $m(e_s) - m(e'_s)$) below a threshold will be discarded. After the first step, we choose additional Ξ pairs with the greatest metric difference to generate our training instances.

The training instances and their label generation is the same as for PRO, except that we also add a global weight (w_G) to each training instance to indicate its importance. In this case, our training instances are:

$$\begin{aligned} &\{+, w_G, h(e_s) - h(e'_s)\} \text{ if } m(e) - m(e') > 0 \\ &\{-, w_G, h(e_s) - h(e'_s)\} \text{ if } m(e) - m(e') < 0 \end{aligned} \quad (1)$$

We use w_G to penalise training samples generated from unreachable sentences. For the dataset in our experiments, empirical results have shown that a translation dataset with a SIMPBLEU score of 0.4 has acceptable translation quality. Therefore, we downweight a training sentence exponentially if the oracle candidate BLEU score is below 0.4. The w_G parameter is defined as:

$$w_G = \begin{cases} 1 & \text{if } BLEU_{Top} \geq 0.4 \\ e^{BLEU_{Top} - 0.4} & \text{if } BLEU_{Top} < 0.4 \end{cases} \quad (2)$$

where the $BLEU_{Top}$ is the BLEU score of the oracle candidate.

After the sampling and training instance generation, we optimise the weights by any off-the-shelf binary classifier that supports weighted training instances. In our experiment, we use the MegaM (Daume, 2004) classifier, the same classifier as in PRO.

5 Results

For our primary submission, we used the two parts of the newsdev2016 development set in two ways: the first half (named newsdev_1) was used to tune our phrase-based SMT system, while the second half (named newsdev_2) was used as an internal test set. The results on the official newstest2016 corpus are presented in Table 2.

[7] A reachable sentence is a sentence for which the model can produce exactly the reference. However, exactly reproducing the reference is not always possible. Therefore in this paper we define a reachable sentence as the best translation hypothesis of a given sentence to reach a certain score.

Algorithm 1 Weighted Ranking Optimisation

Require: Development corpus $D = (f^t, r^t)_{s=1}^{S}$, Initial random weights Λ_0, $\Gamma = 5000$, $\Xi = 50$

for $i = 1$ to I iterations **do**
  MegaM Training instances $R = \{\}$
  for each (f, e) in D **do**
    Calculate w_G acc. Eq. 2
    $r_s = \{\}$
    $Nb =$ DecodeNbest(Λ_i, f)
    $Nb_{top} = 10\%$ best SIMPBLEU(Nb)
    while length(r_s) $< \Gamma$ **do**
      select e_s from Nb_{top}
      select e'_s from Nb
      if $|m(e_s,) - m(e'_s,)| >$ threshold **then**
        Generate samples x acc. Eq. 1
        $r_s \leftarrow r_s + x$
      end if
    end while
    Sort s according to $|m(e_s,) - m(e'_s,)|$
    $R \leftarrow \Xi$ samples with the largest BLEU difference in r_s
  end for
  $\Lambda_{i+1} \leftarrow$ MegaM(R)
end for
return (Λ_{i+1}, R)

We can observe that n-best rescoring with additional features from QE can help identify better hypotheses within the pool of translation candidates. However, as we can see in last the row, we are still far from selecting the best possible hypothesis among those in the n-best list. This "oracle" selection corresponds to the upper bound performance using the current n-best list, based on Meteor (Denkowski and Lavie, 2011) scores measured for each translation candidate against its reference translation. This allows us to compare the actual rank of a translation hypothesis after the rescoring process with the rank it should theoretically have, if our rescoring method were perfect. We also noticed that most of the weights associated with the QE features are set to 0 after the training of the rescoring weights, and therefore most of these features do not get used.

Table 3 shows the performances of our phrase-based system tuned with either PRO or WRO, instead of MERT. We ran these two tuning algorithms on two different development sets: first on newsdev_2, similarly to our rescoring system, second using all the three development sets available for the task combined. We observe that with a smaller development set WRO performs similarly to our system tuned with PRO. However, when the size of the tuning corpus increases, PRO is able to benefit more from the latter, while the system tuned with WRO does not improve its performance.

	BLEU	BLEU-c	TER
MERT	24.17	**23.63**	80.13
+ rescored n-best	**24.49**	23.25	**78**
Oracle	34.56	32.81	69.54

Table 2: BLEU, BLEU-Cased and Translation Error Rate (TER) scores on newstest2016 of our phrase-based SMT submission with and without the use of n-best rescoring. The third line shows the upper bound of our system with the n-best entries scored and sorted against the reference translations using Meteor. The improvement in BLEU for our n-best rescoring over the baseline MERT is statistically significant with $p \leq 0.05$.

6 Conclusions

We presented our phrase-based MT system built using Moses and two variants of this system that were submitted to the WMT16 English-Romanian translation task. As a primary system, we used n-best rescoring with QE features in an attempt to help identify the best translation hypothesis within a 1000 distinct n-best list. We observed some improvements from rescoring, but also the fact that some of the QE features had weights set to zero, and therefore were not used. In future work, we will experiment with a larger QE feature set, which could help us identify more useful features.

As a secondary system, we submitted the phrase-based system trained with WRO, an optimisation algorithm based on PRO which targets weaknesses of PRO in sampling translation candidates. The two algorithms performed similarly on the task, with PRO obtaining better results from using larger development sets.

Acknowledgments

This work was supported by the QT21 (H2020 No. 645452) project.

Algorithm	BLEU	BLEU-c	TER
Dev set: newsdev_2			
WRO	**24.64***	**23.35***	**76.29**
PRO	24.58	23.30	76.30
Dev set: newdev_1 + newsdev_2 + setimes2			
WRO	24.63	23.36	77.20
PRO	**24.76**	**23.49**	**77.05**

Table 3: BLEU, BLEU-Cased and Translation Error Rate (TER) scores of our phrase-based SMT submission on newstest2016 and tuned either with WRO or PRO. In the first row, we only used newsdev_2 as dev set, while in the second row we concatenated all the three dev sets together. The * indicates that the observed improvement of WRO over PRO are statistically significant with $p \leq 0.05$.

References

Alexandre Allauzen, Lauriane Aufrant, Franck Burlot, Elena Knyazeva, Thomas Lavergne, and François Yvon. 2016. LIMSI@WMT'16 : Machine translation of news. Berlin, Germany, August.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 10–51, Montréal, Canada, June. Association for Computational Linguistics.

Colin Cherry and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 427–436. Association for Computational Linguistics.

Hal Daume. 2004. Notes on cg and lm-bfgs optimization of logistic regression. *Unpublished*.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 85–91. Association for Computational Linguistics.

Qin Gao and Stephan Vogel. 2008. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57. Association for Computational Linguistics.

Kenneth Heafield. 2011. KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland, United Kingdom, July.

Mark Hopkins and Jonathan May. 2011. Tuning as ranking. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1352–1362, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantine, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. pages 177–180, Prague, Czech Republic, June.

Matouš Macháček and Ondřej Bojar. 2013. Results of the WMT13 metrics shared task. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 45–51, Sofia, Bulgaria, August. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. pages 160–167, Sapporo, Japan, July.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July.

Anthony Rousseau. 2013. Xenc: An open-source tool for data selection in natural language processing. *The Prague Bulletin of Mathematical Linguistics*, (100):73–82.

Xingyi Song, Trevor Cohn, and Lucia Specia. 2013. Bleu deconstructed: Designing a better mt evaluation metric. In *CICLING*.

Lucia Specia, Gustavo Paetzold, and Carolina Scarton. 2015. Multi-level translation quality prediction with quest++. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 115–120, Beijing, China, July. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.