The QMUL System Description for IWSLT 2010

Sirvan Yahyaei

School of Electronic Engineering and Computer Science
Queen Mary, University of London
London E1 4NS, UK
sirvan@eecs.qmul.ac.uk

Christof Monz

ISLA, Informatics Institute
University of Amsterdam, Science Park 107
1098 XG Amsterdam, The Netherlands
c.monz@uva.nl

Abstract

The QMUL submission to IWSLT 2010 is a phrase-based statistical MT system. A multi-stack, multi-beam decoder with several features, with weights tuned on the provided development data through Minimum Error Rate Training (MERT) algorithm. This year QMUL participated in Arabic-English, French-English and Turkish-English language pairs of the BTEC task.

A discriminative reordering model is added as a feature to improve the reordering capabilities of the decoder. In addition, an algorithm is devised to determine the best distortion limit for each hypothesis expansion. Improvements in quality were also gained by different means in different stages of the training and decoding.

1. Introduction

QMUL submitted runs at IWSLT 2010 evaluation campaign for all the three language pairs of BTEC task. This paper reports the technical details of the system used to perform the translation and the particular improvements of the baseline system to make our submission more competitive.

Our main focus in this submission was on improving the reordering capabilities of the decoder, however, improvements were gained by experimenting with different word-alignment strategies and dealing with out of vocabulary (OOV) words.

The training data provided for the IWSLT BTEC task is relatively small and since the sentences are transcripts of conversations, most of them are very short. This enabled us, to perform the cycle of training, tuning and testing more frequently and investigate many small features and changes. A few of the modifications helped the translation performance, while most of them had insignificant impact.

In Section 2 we describe the baseline system used for the three translation tasks. Section 3 explains our reordering model. In Section 4 results of the baseline and the reordering models are reported and finally, Section 5 concludes the paper.

2. Baseline System

2.1. Preprocessing

For the Arabic-English task, we removed all the diacritics from the Arabic side and normalised the numbers and the punctuations. Buckwalter's morphological analyser is used to tokenise the Arabic side and a simple English tokeniser and lower-caser for the English side.

For French-English pair, we used a simple tokeniser, which works for all European languages in addition to lower-casing both sides. It separates most of the words by whitespace and punctuation characters, but keeps a few exceptions based on a manually created list.

For Turkish-English pair, we used Morfessor [1] to tokenise the Turkish side. Morfessor finds segmentation of the words in an unsupervised manner. The Turkish side of the bitext and all the development data are fed into the Morfessor algorithm to produce segmentations for words which often are similar to linguistic morphemes. Morfessor divides words into multiple morphs including prefixes, stems and suffixes. We retain all the morphs and separate them by a whitespace. We avoided using other publicly available Turkish morphological analysers, since they were using extra training data. We lower-cased both sides of this language pair. Table 1 shows the effect of the preprocessing step on the vocabulary size of the data sets.

2.2. Out-of-Vocabulary Words

For a small size training data such as the one provided, unknown words are a significant problem. Intuitively, many of the unknown words are morphological variations of known words, particularly for morphologically rich languages such as Arabic and Turkish. Therefore, we used simple stemming algorithms to find matches of the unknown words. We search to find a match for the unknown word in the test data among the stemmed words in the training data, then we look for finding a match for the stemmed version of the unknown words in the original training data. Finally, the search is done to find a match of the stemmed unknown words in the stemmed training data. For any match found, the unknown word is replaced with the unstemmed word in the training data. Ta-

	Arabic	French	Turkish	English
Tokens w/o tokenisation	159k	160k	112k	153k
Tokens w tokenisation	170k	200k	162k	189k
Vocabulary w/o tokenisation	37516	35799	39545	32619
Vocabulary w tokenisation	14519	9212	6098	7182
Singletons w/o tokenisation	29852	28572	32410	26444
Singletons w tokenisation	7426	4232	711	3116

Table 1: The effect of preprocessing on the number of tokens and the vocabulary size for all three language pairs. Singletons are words that occur once in the collection.

Data set	Source language	Words	Vocabulary	OOV before	OOV after
IWSLT03.ar-en	Arabic	3323	1095	111	64
IWSLT04.ar-en	Arabic	3479	1189	101	47
IWSLT05.ar-en	Arabic	3375	1182	124	56
IWSLT07.ar-en	Arabic	3158	1100	165	78
IWSLT08.ar-en	Arabic	3414	1130	153	77
IWSLT09.ar-en	Arabic	3135	1039	155	82
IWSLT10.ar-en	Arabic	3207	1096	127	54
IWSLT03.fr-en	French	4063	957	92	69
IWSLT04.fr-en	French	4068	1026	85	52
IWSLT05.fr-en	French	4052	994	89	65
IWSLT09.fr-en	French	3877	888	70	45
IWSLT10.fr-en	French	3813	901	61	43
IWSLT03.tr-en	Turkish	3131	1142	152	86
IWSLT04.tr-en	Turkish	3096	1209	175	89
IWSLT09.tr-en	Turkish	2944	1071	137	79
IWSLT10.tr-en	Turkish	2910	1102	125	76

Table 2: Number of OOV tokens in the development set before finding replacements and after.

ble 2 shows the number of OOV tokens before and after the replacement.

2.3. Decoder

Our decoder is an in-house built multi-beam, multi-stack phrase-based decoder with most of its functionality based on [2]. The features of the baseline include:

- phrase translation probabilities and lexical probabilities for both directions. The word alignment models were produced using Berkeley Aligner [3]. For all three language pairs, we ran IBM model 1, IBM model 2 and HMM jointly for 5 iterations.
- a 4-gram language model. SRILM toolkit [4], was used to build a 4-gram language model, which includes all 4-grams. SRILM by default excludes 4-grams that occur only once in the training data. Preliminary experiments showed that including them improves the quality of the translation for the three language pairs.
- phrase and word penalties.
- distance-based re-ordering penalty.

There are two distortion parameters in our decoder. The distortion limit, which determines the window size of the reordering and the distortion constraint, which controls the decoder movement mainly based on the first uncovered position. Figures 1 and 2 show the BLEU score for different values of the distortion limit for Arabic-English and French-English. The best distortion limit for Arabic in average is 13, which is not the best performing on the tuning data. In other words, the best distortion limit chosen based on the tuning data is not the best for the testing data. Figure 3 shows the BLEU score for Turkish-English with two different distortion constraints. One is the so-called "Window" constraint [5] and the other is called "Max distortion" [6]. The window constraint restricts the decoder by not letting it choose a phrase with more than dl words away from the first open position of the source sentence, while the max distortion constraint is relaxed about the first open position and only restricts the decoder to select the next phrase in a window of length $2 \times dl$. For all experiments the future distortion cost was also estimated and showed to be crucial, particularly for long distance reorderings.

The decoder was tuned using Minimum Error Rate Training [7], implemented in ZMERT [8] to maximise BLEU [9].

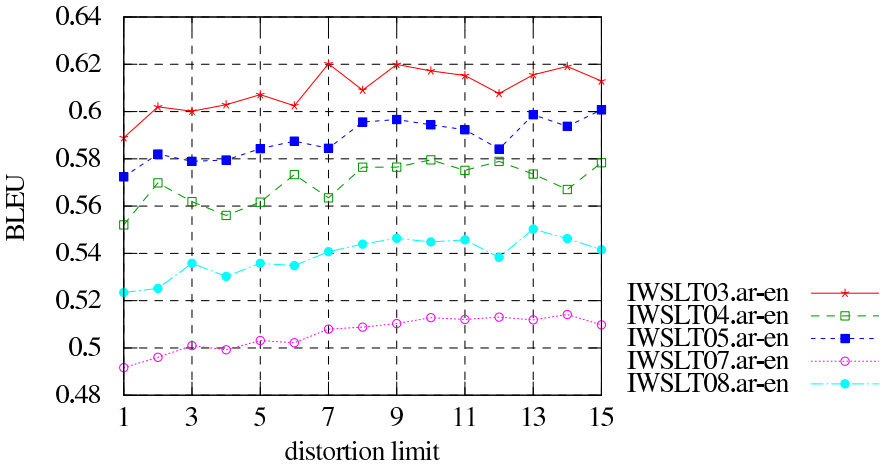

Figure 2: BLEU score changes with different distortion limit values for Arabic-English language pair. The graph in the bottom (IWSLT08) is used for tuning and the rest for testing.

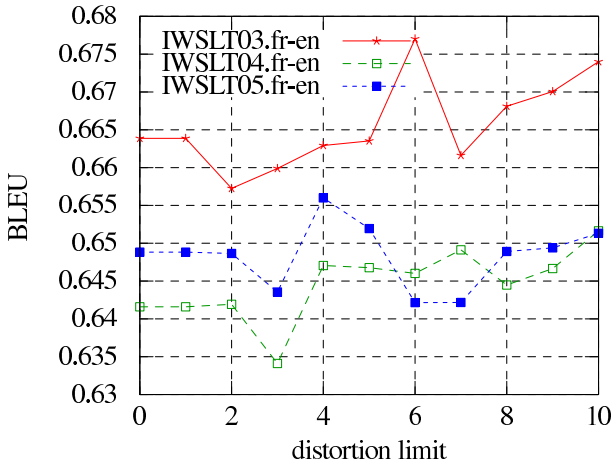

Figure 1: BLEU score changes with different distortion limit values for French-English language pair. IWSLT03 is used for tuning and the rest for testing.

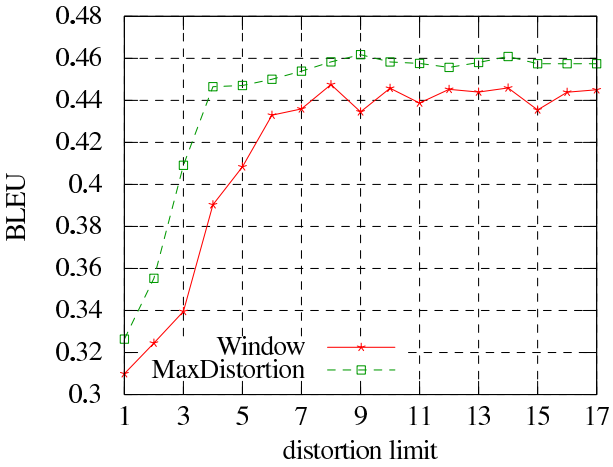

Figure 3: BLEU scores of different distortion limit values for Turkish-English language pair for two distortion constraint.

2.4. Post-Processing

The final output of the decoder was generated through Minimum Bayes Risk Decoding [10], which produced a small, but consistent improvement for all the language pairs. We built a true-caser language model based on the target side of the training data to predict the words that need to be cased. In addition, a detokeniser is used to reverse the tokenisation process.

3. Reordering Model

For this year, we focused on improving the reordering capabilities of the decoder. Turkish is a SOV language and is very different in word order from English. Arabic also requires a substantial amount of middle range reorderings. We built a model to score different reordering decisions based on lexicalised and syntactic features. In addition, we used this model to guide the decoder to dynamically change the size of the reordering window according to the state of the translation. By dynamically adjusting the distortion limit we intended to improve the quality by considering long distance reordering and also avoid noise in situations that reordering is not necessary.

3.1. Discriminative Reordering Model

We built a reordering model for adjusting the distortion limit that takes into account many features from the source sentence. The model is a maximum entropy classifier that predicts the length of the next jump based on the local and global features. The jumps are divided into classes to increase the classification accuracy. For example, jumps with length 2 to 4 are in one class, those with length 5 to 9 in another, etc.

The set of features that we used for the reordering model include lexicalised words, POS-tags, chunks and sentence type. Features for a jump from j to j' in a sentence f_1^J are:

- f_j, $f_{j'}$, $f_j + f_{j'}$
- all the words between j and j'
- part of speech tags of the above words: POS(f_j), POS(f'_j), ...
- bigrams: $f_{j-1} + f_j$ and $f_{j'} + f_{j'+1}$
- bigram part of speech tags of j, j' and the words between them.
- a binary feature indicating that both j and j' are in the same syntactic chunk or not?
- binary feature indicating that f_1^J contains a question mark or not?
- is there a question mark or full stop between j and j'?
- is there a punctuation mark between j and j'?

For Arabic-English and French-English tasks we used all the above features, but for Turkish-English, since we used Morfessor to tokenise the turkish side, the part of speech and chunking features were excluded.

The classifier was optimised by the L-BFGS method [11], implemented in MALLET [12]. To prevent over-fitting, L_1 regularisation was used to reduce the complexity of the model, however, lower translation performance was achieved by using the regularisation. The regularisation can be viewed as a method to select important features and it improves the classification performance of the reordering model in our experiments, but it leads to the translation performance loss at the end.

3.2. Dynamic Distortion

Both translation quality and decoding speed are influenced by changing the distortion limit parameter. The discriminative model built in the previous section, provides us with some information about the reordering needs of a sentence before starting to decode it. We used this information to adjust the distortion limit for each hypothesis expansion.

To find the optimum distortion limit for each position of the source sentence, we use the classifier described in the previous section to compute the probability of jumps from that source position to every other position. The jumps are scored based on these probabilities and used to find the best distortion distance from this position.

To score the jumps after each source position j in the sentence f_1^J, the following equation is used:

$$s_j(j') = \prod_{j''=0}^{j''=j'} p(d_{j,j''}|f_1^J, j, j'') \times \times \prod_{j''=j'+1}^{j''=J+1} (1 - p(d_{j,j''}|f_1^J, j, j'')) \quad (1)$$

$p(d_{j,j'}|f_1^J, j, j')$ is the probability obtained from the classifier for a jump from j to j' with a length in class d and D is the set of jump classes. Equation 1, estimates a score for position j' based on two components: Firstly, the overall probability of jump from j to each position before j'. Secondly, the overall probability of not jumping longer than j' from j. In other words, $s_j(j')$ is the probability the next jump being in the j, j' window.

The final distortion limit estimated by this approach for position j equals to:

$$dl(j) = \text{distance}(j, \arg\max_{j'}\{s_j(j')\}) \quad (2)$$

Changing the distortion limit for each each hypothesis expansion, has the advantage that removes the need for tuning the system with many different distortion limit settings to find the best one. Also, the limit can be very long for some sentences or some parts of a sentence. Changing it for each hypothesis expansion can compensate for long distortion in terms of decoding speed.

4. Experiments

To find the best setting to translate the final test files, we tune the system on different data sets and tested it on the rest of the data sets and chose the data set for tuning with more consistent improvements. Tables 3, 4 and 5 show results for baseline alone, with the OOV replacements and with the dynamic distortion method. No post-processing, as defined in Section 2.4, was applied for the results of the dev data, hence, BLEU scores are calculated on the unprocessed output of the decoder.

To evaluate the contribution of each feature in the classification performance of the discriminative reordering model, we started with the lexical features of f_j and f'_j and added all the features described in Section 3.1 one by one. The most substantial improvements achieved by adding the following features:

- all the words between j and j', which is a binary feature indicating the presence of a word between j and j' or not.

- $f_j + f_{j'}$, which indicates the occurrence of f_j and f'_j together.
- bigram part of speech tags of f_j, f'_j and the words between them. For example, POS(f_{j-1})+POS(f_j)

As mentioned before, the part of speech and chunk features were only used in building the models for Arabic-English and French-English language pairs. For Turkish-English, we only used features that did not require part of speech and chunking information.

PRIMARY runs are the baseline with the dynamic distortion method, replacements of the unknown words and post-processing.

SET	RUN	BLEU
IWSLT08(dev)	BASELINE	**0.5821**
	+OOV-REP	0.5751
	+DYNAMIC-DL	0.5754
IWSLT04(test)	BASELINE	0.5993
	+OOV-REP	0.5982
	+DYNAMIC-DL	**0.6018**
IWSLT05(test)	BASELINE	0.6133
	+OOV-REP	0.6157
	+DYNAMIC-DL	**0.6187**
IWSLT07(test)	BASELINE	**0.5383**
	+OOV-REP	0.5357
	+DYNAMIC-DL	0.5351
IWSLT09(test)	PRIMARY	0.5276
IWSLT10(test)	PRIMARY	0.4425

Table 3: BLEU scores on Arabic-English data sets. OOV-REP is the baseline with some of the unknown words replaced by the matched known word. DYNAMIC-DL is the baseline with the discriminative reordering model and the dynamic distortion method.

5. Conclusion

We built a reordering model and dynamically adjusted the distortion model successfully on the small data sets of IWSLT BTEC task. Although, French and English are very similar in word order, the reordering method improved the translation quality.

In some of the experiments, the BLEU score decreased after replacing the unknown words with the stemmed matched known words. However, by manually checking the matches, most of the them were good replacements and contributed to the meaning of the sentence, therefore, we included this feature for the final tests.

6. Acknowledgements

This work has been funded in part by the European Commission through the CoSyne project FP7-ICT-4-248531 and the GALATEAS project CIP-ICT PSP-2009-3-250430.

SET	RUN	BLEU
IWSLT03(dev)	BASELINE	0.6860
	+OOV-REP	0.6834
	+DYNAMIC-DL	**0.6874**
IWSLT04(test)	BASELINE	0.6605
	+OOV-REP	0.6630
	+DYNAMIC-DL	**0.6694**
IWSLT05(test)	BASELINE	0.6650
	+OOV-REP	0.6600
	+DYNAMIC-DL	**0.6668**
IWSLT09(test)	PRIMARY	0.6180
IWSLT10(test)	PRIMARY	0.5362

Table 4: BLEU scores on French-English data sets. OOV-REP is the baseline with some of the unknown words replaced by the matched known word. DYNAMIC-DL is the baseline with the discriminative reordering model and the dynamic distortion method.

SET	RUN	BLEU
IWSLT03(dev)	BASELINE	0.4783
	+OOV-REP	0.4797
	+DYNAMIC-DL	**0.4814**
IWSLT04(test)	BASELINE	0.4507
	+OOV-REP	0.4505
	+DYNAMIC-DL	**0.4577**
IWSLT09(test)	PRIMARY	0.5354
IWSLT10(test)	PRIMARY	0.5128

Table 5: BLEU scores on Turkish-English data sets. OOV-REP is the baseline with some of the unknown words replaced by the matched known word. DYNAMIC-DL is the baseline with the discriminative reordering model and the dynamic distortion method.

7. References

[1] M. Creutz and K. Lagus, "Unsupervised morpheme segmentation and morphology induction from text corpora using morfessor 1.0," in *Publications in Computer and Information Science, Report A81*, March 2005.

[2] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 48–54.

[3] P. Liang, B. Taskar, and D. Klein, "Alignment by agreement," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2006, pp. 104–111.

[4] A. Stolcke, "SRILM–an extensible language modeling toolkit," in *ICSLP '02: Proceedings of 7th International Conference on Spoken Language Processing*, vol. 2, Denver, USA, 2002, pp. 901–904.

[5] A. Lopez, "Translation as weighted deduction," in *EACL '09: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2009, pp. 532–540.

[6] R. C. Moore and C. Quirk, "Faster beam-search decoding for phrasal statistical machine translation," in *Proceedings of MT Summit XI, European Association for Machine Translation*, Copenhagen, Denmark, September 2007.

[7] F. J. Och, "Minimum error rate training in statistical machine translation," in *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 160–167.

[8] O. F. Zaidan, "Z-MERT: A fully configurable open source tool for minimum error rate training of machine translation systems," *The Prague Bulletin of Mathematical Linguistics*, vol. 91, pp. 79–88, 2009.

[9] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2001, pp. 311–318.

[10] S. Kumar and W. Byrne, "Minimum bayes-risk decoding for statistical machine translation," in *HLT-NAACL 2004: Main Proceedings*, D. M. Susan Dumais and S. Roukos, Eds. Boston, Massachusetts, USA: Association for Computational Linguistics, May 2 - May 7 2004, pp. 169–176.

[11] J. Nocedal, "Updating quasi-newton matrices with limited storage," *Mathematics of Computation*, vol. 35, no. 151, pp. 773–782, 1980.

[12] A. K. McCallum, "MALLET: A machine learning for language toolkit," 2002, http://mallet.cs.umass.edu.