Report of the Shared Task on Learning Reordering from Word Alignments at RSMT 2012

Mitesh M. Khapra[1] *Ananthakrishnan Ramanathan*[1]
Karthik Visweswariah[1]
(1) IBM Research India
{mikhapra,aramana5,v-karthik}@in.ibm.com

Abstract
Several studies have shown that the task of reordering source sentences to match the target order is crucial to improve the performance of Statistical Machine Translation, especially when the source and target languages have significantly divergent grammatical structures. In fact, it is now become a standard practice to include reordering as a pre-processing step or as an integrated module (within the decoder). However, despite the importance of this sub-task, there is no forum dedicated for its evaluation. The objective of this Shared Task was to provide a common benchmarking platform to evaluate state of the art approaches for reordering.

Keywords: Reordering, Machine Translation.

Proceedings of the Workshop on Reordering for Statistical Machine Translation, pages 9–16,
COLING 2012, Mumbai, December 2012.

1 Task Description

The task was to develop a reordering engine to reorder source English sentences to match the order of the target language. For example, the English (SVO language) sentence "Ram drinks water" is translated into Hindi (SOV language) as "Ram paanii piitaa hai (Ram water drinks)". Thus, the correct reordering of this English sentence which matches the target (Hindi) order is "Ram water drinks". The task organizers released high-quality word-alignments (annotated by hand) between English and 3 languages (*viz.*, Urdu, Farsi and Italian). The participants used this training and development data to develop a reordering engine for the mentioned source target language pairs. At evaluation time, a list of source sentences was provided on which the participants had to run their systems and submit the best reordering for each sentence as output by their system. For every language pair, the participants were supposed to submit at least one run which uses only the data provided by the task organizers. This was called a "standard" run. Participants were allowed to submit more than one standard run. In addition, participants were also allowed to submit several "non-standard" runs for each language pair which use data other than that provided by the task organizers.

2 Data

The following data files were provided to the participants for each language pair.

src_tgt.src.[trn|dev].conll : This file is in the standard CoNLL-X format with one word per line and a blank line separating two sentences. Some of the columns have been redefined to suit the reordering task. The columns are as follows:

1. Original index: The index of the word in the original unreordered source sentence
2. word : The lexical form of the word
3. empty : dummy column
4. CPOSTAG: Coarse-grained part-of-speech tag (tagset depends on the language).
5. POSTAG: Fine-grained part-of-speech tag (tagset depends on the language).
6. empty: dummy column
7. Previous Index: The index of the word which precedes this word in the reordered source sentence
8. empty : dummy column
9. empty : dummy column
10. empty : dummy column

Note that the words in the source sentence which do not align to any word in the target sentence will be dropped from the conll file. For example, if the source sentence is "I am going home" and if the word "a" is not aligned to any word in the target sentence then this word will be dropped from the conll file as shown below:

1	I	-	P	PRP	-	0	-	-	-
2	going	-	V	VBG	-	3	-	-	-
3	home	-	N	NOUN	-	1	-	-	-

src_trn.src.[trn|dev].txt : This file contains the complete source sentence (including words which were left unaligned). Example: I am going home.

src_trn.src.[trn|dev].pos : This file contains the pos tags for the complete source sentence (including words which were left unaligned). Example: VRB(I) VMZ(am) VBG(going) NN(home).

src_trn.src.[trn|dev].parse : This file contains a parse for the complete source sentence (including words which were left unaligned). The parse was generated by a state-of-the-art in-house parser.

src_trn.src.[trn|dev].align.info : This file contains indices of only those words which were aligned to some word in the target sentence. Example: 0(I) 2(going) 3 (home)

Note that src_tgt.src.[trn|dev].conll starts at index 1 whereas src_trn.src.[trn|dev].align.info starts at index 0. The participants can use src_tgt.src.[trn|dev].conll and src_trn.src.[trn|dev].align.info to find the words which were left unaligned.

2.1 Language pairs

Table 1 lists the language pairs that were included in the Shared Task and the amount of hand aligned data that was released for each language pair (**En**: English, **Fa**: Farsi, **Ur**: Urdu, **It**: Italian). **This data which was released as a part of the shared task can be obtained by sending a mail to mikhapra@in.ibm.com**.

Language Pair	Train	Dev	Test
En-Fa	5K	500	500
En-Ur	5K	500	500
En-It	4K	500	500

Table 1: Language Pairs included in the Shared Task

3 Evaluation Metrics

The output reorderings were evaluated using two metrics:

- **BLEU** (Papineni et al., 2002): In the past decade, BLEU has been the most widely used metric for MT evaluation. BLEU compares N-grams in the output translation and the reference translation(s), and uses a "brevity penalty" to prevent outputs that are accurate in terms of N-gram match, but too short.

  For reordering, we use the BLEU metric by comparing candidate reorderings with the reference reorderings that we create from the hand-alignments.

  BLEU is calculated as:

$$log(BLEU) = min(1 - \frac{r}{c}, 0) + \sum_{n=1}^{N} \frac{1}{N} log(p_n)$$

$$where, N = 4 \textit{(unigrams, bigrams, trigrams, and 4-grams are matched)}$$

$$r = \textit{length of reference reordering}$$

$$c = \textit{length of candidate reordering}$$

and

$$p_n = \frac{\sum_{C \in Candidates} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}{\sum_{C \in Candidates} \sum_{n\text{-}gram \in C} Count_{clip}(n\text{-}gram)}$$

where C runs over the entire set of candidate reorderings, and $Count_{clip}$ returns the number of n-grams that match in the reference reordering.

- **LRscore** (Birch and Osborne, 2010):

  LRscore was introduced a couple of years ago as a metric to directly measure reordering performance. LRscore uses a distance score in conjunction with BLEU to help evaluate the word order of MT outputs better. Experiments show that this combined metric correlates better with human judgments than BLEU alone (Birch and Osborne, 2010). Since we do not need a lexical metric, we use only the distance metric from LRscore. We will evaluate reordering distance using the following two scores:

  – Hamming distance: This measures the number of disagreements between two permutations:

  $$d_H(\pi, \rho) = 1 - \frac{\sum_{i=1}^{n} x_i}{n}, \text{ where } x_i = \begin{cases} 0 & \text{if } \pi(i) = \rho(i), \\ 1 & \text{otherwise,} \end{cases}$$

  – Kendall's Tau Distance: This measures the minimum number of transpositions of two adjacent symbols necessary to transform one permutation into another:

  $$d_r(\pi, \rho) = 1 - \frac{\sum_{i=1}^{n} \sum_{j=1}^{n} z_{ij}}{Z}, \text{ where } z_{ij} = \begin{cases} 1 & \text{if } \pi(i) < \pi(j) \text{ and } \rho(i) > \rho(j) \\ 0 & \text{otherwise} \end{cases}$$

  $$Z = \frac{(n^2 - n)}{2}$$

  These two distance metrics will be combined with a brevity penalty (as defined in the description of BLEU above).

Links to these evaluation scripts are provided on the workshop webpage[1].

4 Systems

Seven groups requested for the data released in the Shared Task. However, eventually only 3 groups made a clean submission. In this section, we briefly describe the systems submitted by these three groups.

Gupta et al. (2012) treated reordering as translation from unreordered to reordered text. They used a publicly available phrase-based MT toolkit (Moses[2]) for learning this translation model by setting up the unreordered text as the source corpus and the reordered text as

[1] https://sites.google.com/site/rsmt2012/Shared-Task/evaluation-scripts
[2] www.statmt.org/moses/

the target corpus. They experimented with both, a phrase based model and a factor based model. The phrase based model was trained without any preprocessing or reordering of data. The factored based model used 'surface word form' and the 'POS tag' factors as translation-factors for training. The map value <0-0,1> was provided in the training script which indicated a source side (surface) to a target side (surface, POS) mapping. They experimented with different values of distortion-limit and used default settings for all other parameters (for both translation model and language model).

Kunchukuttan and Bhattacharyya (2012) model the problem of reordering source sentences as a problem of reordering word sequences (as opposed to reordering words). They consider source side reordering to be a composition of the following operations on a sentence: (1) Breaking the sentence into word sequences (2) Reversing the words within some word sequences and (3) Reordering the word sequences. They model the first two steps as a sequence labeling problem. The labeling scheme captures word sequence boundaries and reversals, and the training data labels are extracted using the word alignment information provided by the task organizers. The third step is modeled as a Traveling Salesperson (TSP) problem. They consider word sequences, instead of words, to be the cities, and define the cost of traveling from one city to another. The costs are assigned so that the total cost will be minimum for the correct reordering of word sequences. The costs are computed as a function of features of the word sequences involved, and a regression based cost model is learned. The use of word sequences makes solving the TSP problem more tractable, and helps define relevant word-sequence level features for modeling the cost.

Dlougach and Galinskaya (2012) built a syntax-based reordering system using on open-source SMT toolkit (Moses). Using source side parses and word alignment information they learn reordering rules from the small corpus provided by the task organizers. They then apply these rules to reorder the test sentences. They claim that this approach works especially well when source and target languages have different sentence-level order (like Subject-Verb-Object vs. Subject-Object-Verb). Its also accounts for word-level reordering (when nouns are swapped with their corresponding adjectives). While working on this shared task they have also made changes to the source code of Moses, especially its chart decoder. These changes are available in the public repository of Moses (Dlougach and Galinskaya, 2012).

5 Results

As mentioned earlier, the different systems that participated in the Shared Task were evaluated using mBLEU (Table 2), $LR_{Hamming}$ (Table 3) and $LR_{Kendall}$ (Table 4) . To put the results in perspective we compare these systems with a baseline system (which uses no reordering) and a state of the art system which models reordering as a Traveling Salesman Problem (Visweswariah et al., 2011). Note that Visweswariah et al. (2011) did not participate in the Shared Task. Their results are included only for the sake of comparison.

6 Summary and future possibilities

We conducted a Shared Task on Learning Reordering from Word Alignments. The participants were supposed to train reordering models using high quality alignment data as well as pos tagged and parsed source sentences. We provided data for three language pairs (*viz.*, En-Farsi, En-Urdu and En-Italian). A total of seven groups requested for this data but eventually only three groups made a clean submission. These three systems were evaluated

System	En-Fa	En-It	En-Ur
	mBLEU	mBLEU	mBLEU
Baseline	50.0	65.1	38.3
Dlougach and Galinskaya (2012)	65.6	76.7	55.8
Gupta et al. (2012)	55.7	73.0	44.7
Kunchukuttan and Bhattacharyya (2012)	46.4	64.7	37.8
Visweswariah et al. (2011)	68.7	83.0	63.3

Table 2: mBLEU scores of different systems that participated in the Shared Task.

System	En-Fa	En-It	En-Ur
	$LR_{Hamming}$	$LR_{Hamming}$	$LR_{Hamming}$
Baseline	0.418	0.707	0.268
Dlougach and Galinskaya (2012)	0.549	0.771	0.428
Gupta et al. (2012)	0.432	0.751	0.313
Kunchukuttan and Bhattacharyya (2012)	0.086	0.283	0.112
Visweswariah et al. (2011)	0.576	0.817	0.507

Table 3: $LR_{Hamming}$ scores of different systems that participated in the Shared Task.

System	En-Fa	En-It	En-Ur
	$LR_{Kendall}$	$LR_{Kendall}$	$LR_{Kendall}$
Baseline	0.716	0.858	0.491
Dlougach and Galinskaya (2012)	0.748	0.875	0.592
Gupta et al. (2012)	0.712	0.867	0.510
Kunchukuttan and Bhattacharyya (2012)	0.349	0.529	0.348
Visweswariah et al. (2011)	0.764	0.894	0.643

Table 4: $LR_{Kendall}$ scores of different systems that participated in the Shared Task.

using 2 metrics: mBLEU and LR score. Two out of the three participants were able to get reasonable gains over the baseline system (which uses no reordering). The enthusiasm shown for the first offering of this Shared Task was encouraging and we plan to organize this Shared Task again. In the next offering of the Shared Task, we would like to see the performance in the other direction *i.e.* non-English to English.

References

Birch, A. and Osborne, M. (2010). Lrscore for evaluating lexical and reordering quality in mt. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 327–332, Stroudsburg, PA, USA. Association for Computational Linguistics.

Dlougach, J. and Galinskaya, I. (2012). Building a reordering system using tree-to-string hierarchical model. In *Proceedings of the First Workshop on Reordering for Statistical Machine Translation at COLING 2012*, Mumbai, India.

Gupta, R., Patel, R. N., and Shah, R. (2012). Some experiments: Reordering using aligned bilingual corpus. In *Proceedings of the First Workshop on Reordering for Statistical Machine Translation at COLING 2012*, Mumbai, India.

Kunchukuttan, A. and Bhattacharyya, P. (2012). Partially modelling word reordering as a sequence labelling problem. In *Proceedings of the First Workshop on Reordering for Statistical Machine Translation at COLING 2012*, Mumbai, India.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Visweswariah, K., Rajkumar, R., Gandhe, A., Ramanathan, A., and Navratil, J. (2011). A word reordering model for improved machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 486–496, Stroudsburg, PA, USA. Association for Computational Linguistics.