Facilitating Translation Using Source Language Paraphrase Lattices

Jinhua Du, Jie Jiang, Andy Way
CNGL, School of Computing
Dublin City University, Dublin, Ireland
{jdu, jjiang, away}@computing.dcu.ie

Abstract

For resource-limited language pairs, coverage of the test set by the parallel corpus is an important factor that affects translation quality in two respects: 1) out of vocabulary words; 2) the same information in an input sentence can be expressed in different ways, while current phrase-based SMT systems cannot automatically select an alternative way to transfer the same information. Therefore, given limited data, in order to facilitate translation from the input side, this paper proposes a novel method to reduce the translation difficulty using source-side lattice-based paraphrases. We utilise the original phrases from the input sentence and the corresponding paraphrases to build a lattice with estimated weights for each edge to improve translation quality. Compared to the baseline system, our method achieves relative improvements of 7.07%, 6.78% and 3.63% in terms of BLEU score on small, medium and large-scale English-to-Chinese translation tasks respectively. The results show that the proposed method is effective not only for resource-limited language pairs, but also for resource-sufficient pairs to some extent.

1 Introduction

In recent years, statistical MT systems have been easy to develop due to the rapid explosion in data availability, especially parallel data. However, in reality there are still many language pairs which lack parallel data, such as Urdu–English, Chinese–Italian, where large amounts of speakers exist for both languages; of course, the problem is far worse for pairs such as Catalan–Irish. For such resource-limited language pairs, sparse amounts of parallel data would cause the word alignment to be inaccurate, which would in turn lead to an inaccurate phrase alignment, and bad translations would result. Callison-Burch et al. (2006) argue that limited amounts of parallel training data can lead to the problem of low coverage in that many phrases encountered at run-time are not observed in the training data and so their translations will not be learned. Thus, in recent years, research on addressing the problem of unknown words or phrases has become more and more evident for resource-limited language pairs.

Callison-Burch et al. (2006) proposed a novel method which substitutes a paraphrase for an unknown source word or phrase in the input sentence, and then proceeds to use the translation of that paraphrase in the production of the target-language result. Their experiments showed that by translating paraphrases a marked improvement was achieved in coverage and translation quality, especially in the case of unknown words which previously had been left untranslated. However, on a large-scale data set, they did not achieve improvements in terms of automatic evaluation.

Nakov (2008) proposed another way to use paraphrases in SMT. He generates nearly-equivalent syntactic paraphrases of the source-side training sentences, then pairs each paraphrased sentence with the target translation associated with the original sentence in the training data. Essentially, this method generates new training data using paraphrases to train a new model and obtain more useful

Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 420–429,
MIT, Massachusetts, USA, 9-11 October 2010. ©2010 Association for Computational Linguistics

phrase pairs. However, he reported that this method results in bad system performance. By contrast, real improvements can be achieved by merging the phrase tables of the paraphrase model and the original model, giving priority to the latter. Schroeder et al. (2009) presented the use of word lattices for multi-source translation, in which the multiple source input texts are compiled into a compact lattice, over which a single decoding pass is then performed. This lattice-based method achieved positive results across all data conditions.

In this paper, we propose a novel method using paraphrases to facilitate translation, especially for resource-limited languages. Our method does not distinguish unknown words in the input sentence, but uses paraphrases of all possible words and phrases in the source input sentence to build a source-side lattice to provide a diverse and flexible list of source-side candidates to the SMT decoder so that it can search for a best path and deliver the translation with the highest probability. In this case, we neither need to change the phrase table, nor add new features in the log-linear model, nor add new sentences in the training data.

The remainder of this paper is organised as follows. In Section 2, we define the "translation difficulty" from the perspective of the source side, and then examine how well the test set is covered by the phrase table and the parallel training data . Section 3 describes our paraphrase lattice method and discusses how to set the weights for the edges in the lattice network. In Section 4, we report comparative experiments conducted on small, medium and large-scale English-to-Chinese data sets. In Section 5, we analyse the influence of our paraphrase lattice method. Section 6 concludes and gives avenues for future work.

2 What Makes Translation Difficult?

2.1 Translation Difficulty

We use the term "translation difficulty" to explain how difficult it is to translate the source-side sentence in three respects:

- The OOV rates of the source sentences in the test set (Callison-Burch et al., 2006).
- Translatability of a known phrase in the input sentence. Some particular grammatical structures on the source side cannot be directly translated into the corresponding structures on the target side. Nakov (2008) presents an example showing how hard it is to translate an English construction into Spanish. Assume that an English-to-Spanish SMT system has an entry in its phrase table for "*inequality of income*", but not for "*income inequality*". He argues that the latter phrase is hard to translate into Spanish where noun compounds are rare: the correct translation in this case requires a suitable Spanish preposition and a reordering, which are hard for the system to realize properly in the target language (Nakov, 2008).
- Consistency between the reference and the target-side sentence in the training corpus. Nakov (2008) points out that if the target-side sentence in the parallel corpus is inconsistent with the reference of the test set, then in some cases, a test sentence might contain pieces that are equivalent, but syntactically different from the phrases learned in training, which might result in practice in a missed opportunity for a high-quality translation. In this case, if we use paraphrases for these pieces of text, then we might improve the opportunity for the translation to approach the reference, especially in the case where only one reference is available.

2.2 Coverage

As to the first aspect – coverage – we argue that the coverage rate of the new words or unknown words are more and more becoming a "bottleneck" for resource-limited languages. Furthermore, current SMT systems, either phrase-based (Koehn et al., 2003; Chiang, 2005) or syntax-based (Zollmann and Venugopal, 2006), use phrases as the fundamental translation unit, so how much the phrase table and training data can cover the test set is an important factor which influences the translation quality. Table 1 shows the statistics of the coverage of the test set on English-to-Chinese FBIS data, where we can see that the coverage of unigrams is very high, especially when the data is increased to the medium size (200K), where unigram coverage is greater than 90%. Based on the observations of the unknown un-

		20K		Cov.(%)		200K		Cov.(%)	
PL	Tset	PT	Corpus	in PT	in Corpus	PT	Corpus	in PT	in Corpus
1	5,369	3,785	4,704	70.5	87.61	4,941	5,230	92.03	97.41
2	24,564	8,631	15,109	35.14	61.51	16,803	21,071	68.40	85.78
3	37,402	4,538	12,091	12.13	32.33	12,922	22,531	34.55	60.24
4	41,792	1,703	6,150	4.07	14.72	5,974	14,698	14.29	35.17
5	43,008	626	2,933	1.46	6.82	2,579	8,425	5.99	19.59
6	43,054	259	1,459	0.6	3.39	1,192	4,856	2.77	11.28
7	42,601	119	821	0.28	1.93	581	2,936	1.36	6.89
8	41,865	51	505	0.12	1.21	319	1,890	0.76	4.51
9	40,984	34	341	0.08	0.83	233	1,294	0.57	3.16
10	40,002	22	241	0.05	0.6	135	923	0.34	2.31

Table 1: The coverage of the test set by the phrase table and the parallel corpus based on different amount of the training data. "PL" indicates the Phrase Length N, where $\{1 <= N <= 10\}$; "20K" and "200K" represent the sizes of the parallel data for model training and phrase extraction; "Cov." indicates the coverage rate; "Tset" represents the number of unique phrases with the length N in the Test Set; "PT" represents the number of phrases of the Test Set occur in the Phrase Table; "Corpus" indicates the number of phrases of the Test Set appearing in the parallel corpus; "in PT" indicates the coverage of the phrases in the Test Set by the phrase table and correspondingly "in Corpus" represents the coverage of the phrases in the Test Set by the Parallel Corpus.

igrams, we found that most are named entities (NEs) such as person name, location name, etc. From the bigram phrases, the coverage rates begin to significantly decline. It can also be seen that phrases containing more than 5 words rarely appear either in the phrase table or in the parallel corpus, which indicates that data sparseness is severe for long phrases. Even if the size of the corpus is significantly increased (e.g. from 20K to 200K), the coverage of long phrases is still quite low.

With respect to these three aspects of the translation difficulty, especially for data-limited language pairs, we propose a more effective method to make use of the paraphrases to facilitate translation process.

3 Paraphrase Lattice for Input Sentences

In this Section, we propose a novel method to employ paraphrases to reduce the translation difficulty and in so doing increase the translation quality.

3.1 Motivation

Our idea to build a paraphrase lattice for SMT is inspired by the following points:

- Handling unknown words is a challenging issue for SMT, and using paraphrases is an effective way to facilitate this problem (Callison-Burch et al., 2006);

- The method of paraphrase substitution does not show any significant improvement, especially on a large-scale data set in terms of BLEU (Papineni et al., 2002) scores (Callison-Burch et al., 2006);

- Building a paraphrase lattice might provide more translation options to the decoder so that it can flexibly search for the best path.

The major contributions of our method are:

- We consider all N-gram phrases rather than only unknown phrases in the test set, where $\{1 <= N <= 10\}$;

- We utilise lattices rather than simple substitution to facilitate the translation process;

- We propose an empirical weight estimation method to set weights for edges in the word lattice, which is detailed in Section 3.4.

3.2 Paraphrase Acquisition

Paraphrases are alternative ways to express the same or similar meaning given a certain original word, phrase or segment. The paraphrases used in our method are generated from the parallel corpora based on the algorithm in (Bannard and Callison-Burch, 2005), in which paraphrases are identified by pivoting through phrases in another language. In this algorithm, the foreign language translations of an English phrase are identified, all occurrences of those foreign phrases are found, and all English phrases that they translate as are treated as potential paraphrases of the original English phrase (Callison-Burch et al., 2006). A paraphrase has a probability $p(e_2|e_1)$ which is defined as in (2):

$$p(e_2|e_1) = \sum_f p(f|e_1)p(e_2|f) \qquad (1)$$

where the probability $p(f|e_1)$ is the probability that the original English phrase e_1 translates as a particular phrase f in the other language, and $p(e_2|f)$ is the probability that the candidate paraphrase e_2 translates as the foreign language phrase.

$p(e_2|f)$ and $p(f|e_1)$ are defined as the translation probabilities which can be calculated straightforwardly using maximum likelihood estimation by counting how often the phrases e and f are aligned in the parallel corpus as in (2) and (3):

$$p(e_2|f) \approx \frac{count(e_2, f)}{\sum_{e_2} count(e_2, f)} \qquad (2)$$

$$p(f|e_1) \approx \frac{count(f, e_1)}{\sum_f count(f, e_1)} \qquad (3)$$

3.3 Construction of Paraphrase Lattice

To present paraphrase options to the PB-SMT decoder, lattices with paraphrase options are constructed to enrich the source-language sentences. The construction process takes advantage of the correspondence between detected paraphrases and positions of the original words in the input sentence, then creates extra edges in the lattices to allow the decoder to consider paths involving the paraphrase words.

An toy example is illustrated in Figure 1: given a sequence of words $\{w_1, \ldots, w_N\}$ as the input, two phrases $\alpha = \{\alpha_1, \ldots, \alpha_p\}$ and $\beta = \{\beta_1, \ldots, \beta_q\}$ are detected as paraphrases for $S_1 = \{w_x, \ldots, w_y\}$ $(1 \leq x \leq y \leq N)$ and $S_2 = \{w_m, \ldots, w_n\}$ $(1 \leq m \leq n \leq N)$ respectively. The following steps are taken to transform them into word lattices:

1. Transform the original source sentence into word lattices. $N + 1$ nodes $(\theta_k, 0 \leq k \leq N)$ are created, and N edges (referred to as "ORG-E" edges) labeled with w_i $(1 \leq i \leq N)$ are generated to connect them sequentially.

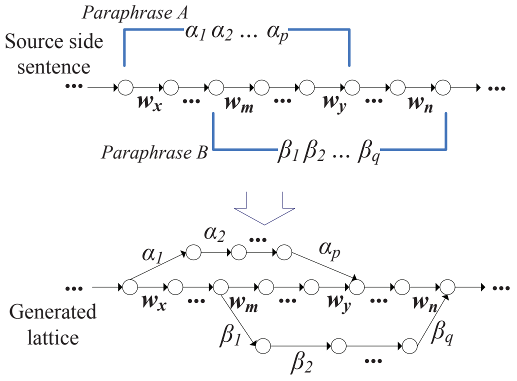

Figure 1: An example of lattice-based paraphrases for an input sentence

2. Generate extra nodes and edges for each of the paraphrases. Taking α as an example, firstly, $p - 1$ nodes are created, and then p edges (referred as "NEW-E" edges) labeled with α_j $(1 \leq j \leq p)$ are generated to connect node θ_{x-1}, $p - 1$ nodes and θ_{y-1}.

Via step 2, word lattices are generated by adding new nodes and edges coming from paraphrases. Note that to build word lattices, paraphrases with multi-words are broken into word sequences, and each of the words produces one extra edge in the word lattices as shown in the bottom part in Figure 1.

Figure 2 shows an example of constructing the word lattice for an input sentence which is from the test set used in our experiments.[1] The top part in Figure 2 represents nodes (double-line circles) and edges (solid lines) that are constructed by the original words from the input sentence, while the bottom part in Figure 2 indicates the final word lattice with the addition of new nodes (single-line circles) and new edges (dashed lines) which come from the paraphrases. We can see that the paraphrase lattice increases the diversity of the source phrases so that it can provide more flexible translation options during the decoding process.

[1] Figure 2 contains paths that are duplicates except for the weights. We plan to handle this in future work.

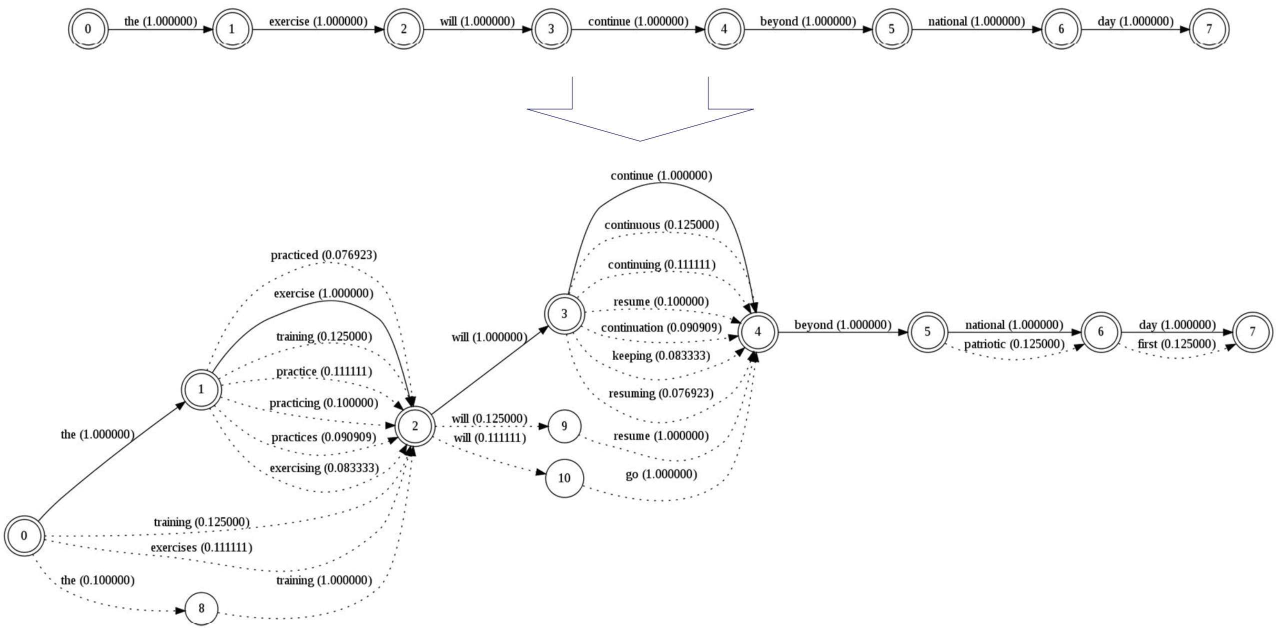

Figure 2: An example of how to build a paraphrase lattice for an input sentence

3.4 Weight Estimation

Estimating and normalising the weight for each edge in the word lattice is a challenging issue when the edges come from different sources. In this section, we propose an empirical method to set the weights for the edges by distinguishing the original ("ORG-E") and new ("NEW-E") edges in the lattices. The aim is to utilize the original sentences as the references to weight the edges from paraphrases, so that decoding paths going through "ORG-E" edges will tend to have higher scores than those which use "NEW-E" ones. The assumption behind this is that the paraphrases are alternatives for the original sentences, so decoding paths going though them ought to be penalised.

Therefore, for all the "ORG-E" edges, their weights in the lattice are set to 1.0 as the reference. Thus, in the log-linear model, decoding paths going though these edges are not penalised because they do not come from the paraphrases.

By contrast, "NEW-E" are divided into two groups for the calculation of weights:

- For "NEW-E" edges which are outgoing edges of the lattice nodes that come from the original sentences, the probabilities $p(e_s|e_i)$[2] of their corresponding paraphrases are utilised to produce empirical weights. Supposing that a set of paraphrases $\mathbf{X} = \{x_1, \ldots, x_k\}$ start at node A which comes from the original sentence, so that $\mathbf{X}$ are sorted descendingly based on the probabilities $p(e_s|e_i)$, their corresponding edges for node A are $\mathbf{G} = \{g_1, \ldots, g_k\}$, then the weights are calculated as in (4):

$$w(e_i) = \frac{1}{k+i} \ (1 <= i <= k) \qquad (4)$$

  where k is a predefined parameter to trade off between decoding speed and the number of potential paraphrases being considered. Thus, once a decoding path goes though one of these edges, it will be penalised according to its paraphrase probabilities.

- For all other "NEW-E" edges, their weights are set to 1.0, because the paraphrase penalty has been counted in their preceding "NEW-E" edges.

Figure 2 illustrates the weight estimation results. Nodes coming from the original sentences are drawn in double-line circles (e.g. nodes 0 to 7), while

[2] e_s indicates the source phrase S, e_i represents one of the paraphrases of S.

nodes created from paraphrases are shown in single-line circles (e.g. nodes 8 to 10). "ORG-E" edges are drawn in solid lines and "NEW-E" edges are shown using dashed lines. As specified previously, "ORG-E" edges are all weighted by 1.0 (e.g. edge labeled "the" from node 0 to 1). By contrast, "NEW-E" edges in the first group are weighted by equation (4) (e.g. edges in dashed lines start from node 0 to node 2 and 8), while others in the second group are weighted by 1.0 (e.g. edge labeled "training" from node 8 to 2). Note that penalties of the paths going through paraphrases are counted by equation (4), which is represented by the weights of "NEW-E" edges in the first group. For example, starting from node 2, paths going to node 9 and 10 are penalised because lattice weights are also considered in the log-linear model. However, other edges do not imply penalties since their weights are set to 1.0.

The reason to set all weights for the "ORG-E" edges to a uniform weight (e.g. 1.0) instead of a lower empirical weight is to avoid excessive penalties for the original words. For example, in Figure 2, the original edge from node 3 to 4 (*continue*) has a weight of 1.0, so the paths going though the original edges from node 2 to 4 (*will continue*) have a higher lattice score ($1.0 \times 1.0 = 1.0$) than the paths going through the edges of paraphrases (e.g. *will resume* (score: $0.125 \times 1.0 = 0.125$) and *will go* (score: $0.11 \times 1.0 = 0.11$)), or any other mixed paths that goes through original edges and paraphrase edges, such as *will continuous* (score: $1.0 \times 0.125 = 0.125$). The point is that we should have more trust when translating the original words, but if we penalise (set weights < 1.0) the "ORG-E" edges whenever there is a paraphrase for them, then when considering the context of the lattice, paraphrases will be favoured systematically. That is why we just penalise the "NEW-E" edges in the first group and set other weights to 1.0.

As to unknown words in the input sentence, even if we give them a prioritised weight, they would be severely penalised in the decoding process. So we do not need to distinguish unknown words when building and weighting the paraphrase lattice.

4 Experiments

4.1 System and Data Preparation

For our experiments, we use Moses (Koehn et al., 2007) as the baseline system which can support lattice decoding. We also realise a paraphrase substitution-based system (Para-Sub)[3] based on the method in (Callison-Burch, 2006) to compare with the baseline system and our proposed paraphrase lattice-based (Lattice) system.

The alignment is carried out by GIZA++ (Och and Ney, 2003) and then we symmetrized the word alignment using the grow-diag-final heuristic. The maximum phrase length is 10 words. Parameter tuning is performed using Minimum Error Rate Training (Och, 2003).

The experiments are conducted on English-to-Chinese translation. In order to fully compare our proposed method with the baseline and the "Para-Sub" system, we perform the experiments on three different sizes of training data: 20K, 200K and 2.1 million pairs of sentences. The former two sizes of data are derived from FBIS,[4] and the latter size of data consists of part of HK parallel corpus,[5] ISI parallel data,[6] other news data and parallel dictionaries from LDC. All the language models are 5-gram which are trained on the monolingual part of parallel data.

The development set (devset) and the test set for experiments using 20K and 200K data sets are randomly extracted from the FBIS data. Each set includes 1,200 sentences and each source sentence has one reference. For the 2.1 million data set, we use a different devset and test set in order to verify whether our proposed method can work on a language pair with sufficient resources. The devset is the NIST 2005 Chinese-English current set which has only one reference for each source sentence and the test set is the NIST 2003 English-to-Chinese current set which contains four references for each source sentence. All results are reported in BLEU and TER (Snover et al., 2006) scores.

[3] We use "Para-Sub" to represent their system in the rest of this paper.

[4] This is a multilingual paragraph-aligned corpus with LDC resource number LDC2003E14.

[5] LDC number: LDC2004T08.

[6] LDC number: LDC2007T09.

SYS	20K				200K			
	BLEU	CI 95%	pair-CI 95%	TER	BLEU	CI 95%	pair-CI 95%	TER
Baseline	14.42	[-0.81, +0.74]	–	75.30	23.60	[-1.03, +0.97]	–	63.56
Para-Sub	14.78	[-0.78, +0.82]	[+0.13, +0.60]	73.75	23.41	[-1.04, +1.00]	[-0.46, +0.09]	63.84
Lattice	**15.44**	[-0.85, +0.84]	[+0.74, +1.30]	**73.06**	**25.20**	[-1.11, +1.15]	[+1.19, +2.01]	**62.37**

Table 2: Comparison between the baseline, "Para-Sub" and our "Lattice" (paraphrase lattice) method.

The paraphrase data set used in our lattice-based and the "Para-Sub" systems is same which is derived from the "Paraphrase Phrase Table"[7] of TER-Plus (Snover et al., 2009). The parameter k in equation 4 is set to 7.

4.2 Paraphrase Filtering

The more edges there are in a lattice, the more complicated the decoding is in the search process. Therefore, in order to reduce the complexity of the lattice and increase decoding speed, we must filter out some potential noise in the paraphrase table. Two measures are taken to optimise the paraphrases when building a paraphrase lattice:

- Firstly, we filter out all the paraphrases whose probability is less than 0.01;

- Secondly, given a source-side input sentence, we retrieve all possible paraphrases and their probabilities for source-side phrases which appear in the paraphrase table. Then we remove the paraphrases which are not occurred in the "phrase table" of the SMT system. This measure intends to avoid adding new "unknown words" to the source-side sentence. After this measure, we can acquire the final paraphrases which can be denoted as a quadruple $< SEN\_ID, Span, Para, Prob >$, where "SEN_ID" indicates the ID of the input sentence, "Span" represents the span of the source-side phrase in the original input sentence, "Para" indicates the paraphrase of the source-side phrase, and "Prob" represents the probability between the source-side phrase and its paraphrase, which is used to set the weight of the edge in the lattice. The quadruple is used to construct the weighted lattice.

4.3 Experimental Results

The experimental results conducted on small and medium-sized data sets are shown in Table 2. The 95% confidence intervals (CI) for BLEU scores are independently computed on each of three systems, while the "pair-CI 95%" are computed relative to the baseline system only for "Para-Sub" and "Lattice" systems. All the significance tests use bootstrap and paired-bootstrap resampling normal approximation methods (Zhang and Vogel, 2004).[8] Improvements are considered to be significant if the left boundary of the confidence interval is larger than zero in terms of the "pair-CI 95%". It can be seen that 1) our "Lattice" system outperforms the baseline by 1.02 and 1.6 absolute (7.07% and 6.78% relative) BLEU points in terms of the 20K and 200K data sets respectively, and our system also decreases the TER scores by 2.24 and 1.19 (2.97% and 1.87% relative) points than the baseline system. In terms of the "pair-CI 95%", the left boundaries for 20K and 200K data are respectively "+0.74" and "+1.19", which indicate that the "Lattice" system is significantly better than the baseline system on these two data sets. 2) The "Para-Sub" system performs slightly better (0.36 absolute BLEU points) than the baseline system on the 20K data set, but slightly worse (0.19 absolute BLEU points) than the baseline on the 200K data set, which indicates that the paraphrase substitution method used in (Callison-Burch et al., 2006) does not work on resource-sufficient data sets. In terms of the "pair-CI 95%", the left boundary for 20K data is "+0.13", which indicates that it is significantly better than the baseline system, while the left boundary is "-0.46" for 200K data, which indicates that the "Para-Sub" system is significantly worse than the baseline system. 3) comparing the "Lattice" system with the "Para-Sub"

[7] `http://www.umiacs.umd.edu/~snover/terp/downloads/terp-pt.v1.tgz`.

[8] `http://projectile.sv.cmu.edu/research/public/tools/bootStrap/tutorial.htm`.

SYS	BLEU	CI 95%	pair-CI 95%	NIST	TER
Baseline	14.04	[-0.73, +0.40]	–	6.50	74.88
Para-Sub	14.13	[-0.56, +0.56]	[-0.18, +0.40]	6.52	74.43
Lattice	**14.55**	[-0.75, +0.32]	[+0.15,+0.83]	**6.55**	**73.28**

Table 3: Comparison between the baseline and our paraphrase lattice method on a large-scale data set.

system, the “pair-CI 95%” for 20K and 200K data are respectively [+0.41, +0.92] and [+1.40, +2.17], which indicates that the “Lattice” system is significantly better than the “Para-Sub” system on these two data sets as well. 4) In terms of the two metrics, our proposed method achieves the best performance, which shows that our method is effective and consistent on different sizes of data.

In order to verify our method on large-scale data, we also perform experiments on 2.1 million sentence-pairs of English-to-Chinese data as described in Section 4.1. The results are shown in Table 3. From Table 3, it can be seen that the “Lattice” system achieves an improvement of 0.51 absolute (3.63% relative) BLEU points and a decrease of 1.6 absolute (2.14% relative) TER points compared to the baseline. In terms of the “pair-CI 95%”, the left boundary for the “Lattice” system is “+0.15” which indicates that it is significantly better than the baseline system in terms of BLEU. Interestingly, in our experiment, the “Para-Sub” system also outperforms the baseline on those three automatic metrics. However, in terms of the “pair-CI 95%”, the left boundary for the “Para-Sub” system is “-0.18” which indicates that it is not significantly better than the baseline system in terms of BLEU. The results also show that our proposed method is effective and consistent even on a large-scale data set.

It also can be seen that the improvement on 2.1 million sentence-pairs is less than that of the 20K and 200K data sets. That is, as the size of the training data increases, the problems of data sparseness decrease, so that the coverage of the test set by the parallel corpus will correspondingly increase. In this case, the role of paraphrases in decoding becomes a little weaker. On the other hand, it might become a kind of noise to interfere with the exact translation of the original source-side phrases when decoding. Therefore, our proposed method may be more appropriate for language pairs with limited resources.

5 Analysis

5.1 Coverage of Paraphrase Test Set

The coverage rate of the test set by the phrase table is an important factor that could influence the translation result, so in this section we examine the characteristics of the updated test set that adds in the paraphrases. We take the 200K data set to examine the coverage issue. Table 4 is an illustration to compare the new coverage and the old coverage (without paraphrases) on medium sized training data.

PL	Tset	PT	New Cov.(%)	Old Cov.(%)
1	9,264	8,994	**97.09**	92.03
2	32,805	25,796	**78.63**	68.40
3	39,918	15,708	**39.35**	34.55
4	42,247	6,479	**15.34**	14.29
5	43,088	2,670	**6.20**	5.99
6	43,066	1,204	**2.80**	2.77
7	42,602	582	1.37	1.36
8	41,865	319	0.76	0.76
9	40,984	233	0.57	0.57
10	40,002	135	0.34	0.34

Table 4: The coverage of the paraphrase-added test set by the phrase table on medium size of the training data.

From Table 4, we can see that the coverage of unigrams, bigrams, trigrams and 4-grams goes up by about 5%, 10%, 5% and 1%, while from 5-grams there is only a slight or no increase in coverage. These results show that 1) most of the paraphrases that are added in are lower-order n-grams; 2) the paraphrases can increase the coverage of the input by handling the unknown words to some extent.

However, we observed that most untranslated words in the “Para-Sub” and “Lattice” systems are still NEs, which shows that in our paraphrase table, there are few paraphrases for the NEs. Therefore, to further improve the translation quality using paraphrases, we also need to acquire the paraphrases for NEs to increase the coverage of unknown words.

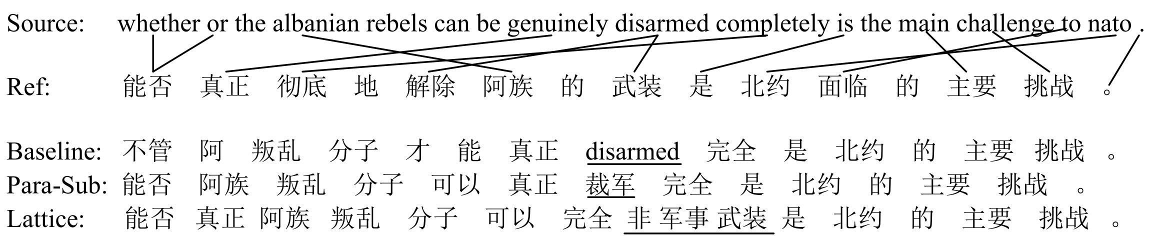
Source: whether or the albanian rebels can be genuinely disarmed completely is the main challenge to nato .
Ref: 能否 真正 彻底 地 解除 阿族 的 武装 是 北约 面临 的 主要 挑战 。
Baseline: 不管 阿 叛乱 分子 才 能 真正 disarmed 完全 是 北约 的 主要 挑战 。
Para-Sub: 能否 阿族 叛乱 分子 可以 真正 裁军 完全 是 北约 的 主要 挑战 。
Lattice: 能否 真正 阿族 叛乱 分子 可以 完全 非 军事 武装 是 北约 的 主要 挑战 。

Figure 3: An example from three systems to compare the processing of OOVs

5.2 Analysis on Translation Results

In this section, we give an example to show the effectiveness of using paraphrase lattices to deal with unknown words. The example is evaluated according to both automatic evaluation and human evaluation at sentence level.

See Figure 3 as an illustration of how the paraphrase-based systems process unknown words. According to the word alignments between the source-side sentence and the reference, the word "disarmed" is translated into two Chinese words "解除" and "武装". These two Chinese words are discontinuous in the reference, so it is difficult for the PB-SMT system to correctly translate the single English word into a discontinuous Chinese phrase. In fact in this example, "disarmed" is an unknown word and it is kept untranslated in the result of the baseline system. In the "Para-Sub" system, it is translated into "裁军" based on a paraphrase pair PP_1 = "disarmed || disarmament || 0.087" and its translation pair T_1 = "disarmament || 裁军". The number "0.087" is the probability p_1 that indicates to what extent these two words are paraphrases. It can be seen that although "裁军" is quite different from the meaning of "disarmed", it is understandable for human in some sense. In the "Lattice" system, the word "disarmed" is translated into three Chinese words "非 军事 武装" based on a paraphrase pair PP_2 = "disarmed || demilitarized || 0.099" and its translation pair T_2 = "demilitarized || 非 军事 武装". The probability p_2 is slightly greater than p_1.

We argue that the reason that the "Lattice" system selects PP_2 and T_2 rather than PP_1 and T_1 is because of the weight estimation in the lattice. That is, PP_2 is more prioritised, while PP_1 is more penalised based on equation (4).

From the viewpoint of human evaluation, the paraphrase pair PP_2 is more appropriate than PP_1, and the translation T_2 is more similar to the original meaning than T_1. The sentence-level automatic evaluation scores for this example in terms of BLEU and TER metrics are shown in Table 5.

SYS	BLEU	TER
Baseline	20.33	66.67
Para-Sub	21.78	**53.33**
Lattice	**23.51**	**53.33**

Table 5: Comparison on sentence-level scores in terms of BLEU and TER metrics.

The BLEU score of the "Lattice" system is much higher than the baseline, and the TER score is quite a bit lower than the baseline. Therefore, from the viewpoint of automatic evaluation, the translation from the "Lattice" system is also better than those from the baseline and "Para-Sub" systems.

6 Conclusions and Future Work

In this paper, we proposed a novel method using paraphrase lattices to facilitate the translation process in SMT. Given an input sentence, our method firstly discovers all possible paraphrases from a paraphrase database for N-grams ($1 <= N <= 10$) in the test set, and then filters out the paraphrases which do not appear in the phrase table in order to avoid adding new unknown words on the input side. We then use the original words and the paraphrases to build a word lattice, and set the weights to prioritise the original edges and penalise the paraphrase edges. Finally, we import the lattice into the decoder to perform lattice decoding. The experiments are conducted on English-to-Chinese translation using the FBIS data set with small and medium-sized amounts of data, and on a large-scale corpus of 2.1

million sentence pairs. We also performed comparative experiments for the baseline, the "Para-Sub" system and our paraphrase lattice-based system. The experimental results show that our proposed system significantly outperforms the baseline and the "Para-Sub" system, and the effectiveness is consistent on the small, medium and large-scale data sets.

As for future work, firstly we plan to propose a pruning algorithm for the duplicate paths in the lattice, which will track the edge generation with respect to the path span, and thus eliminate duplicate paths. Secondly, we plan to experiment with another feature function in the log-linear model to discount words derived from paraphrases, and use MERT to assign an appropriate weight to this feature function.

Acknowledgments

Many thanks to the reviewers for their insightful comments and suggestions. This work is supported by Science Foundation Ireland (Grant No. 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Dublin City University.

References

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *43rd Annual meeting of the Association for Computational Linguistics*, Ann Arbor, MI, pages 597–604.

Chris Callison-Burch, Philipp Koehn and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of HLT-NAACL 2006: Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL*, NY, USA, pages 17–24.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *43rd Annual meeting of the Association for Computational Linguistics*, Ann Arbor, MI, pages 263–270.

Philipp Koehn, Franz Josef Och and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT-NAACL 2003: conference combining Human Language Technology conference series and the North American Chapter of the Association for Computational Linguistics conference series*, Edmonton, Canada, pages 48–54.

Philipp Koehn, Hieu Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, Wade Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *ACL 2007: demo and poster sessions*, Prague, Czech Republic, pages 177–180.

Preslav Nakov. 2008. Improving English-Spanish statistical machine translation: experiments in domain adaptation, sentence paraphrasing, tokenization, and recasing. In *Proceedings of ACL-08:HLT. Third Workshop on Statistical Machine Translation*, Columbus, Ohio, USA, pages 147–150.

Franz Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *41st Annual meeting of the Association for Computational Linguistics*, Sapporo, Japan, pages 160–167.

Franz Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *40th Annual meeting of the Association for Computational Linguistics*, Philadelphia, PA, pages 311–318.

Josh Schroeder, Trevor Cohn and Philipp Koehn. 2009. Word Lattices for Multi-source Translation. In *Proceedings of the 12th Conference of the European Chapter of the ACL*, Athens, Greece, pages 719–727.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, Cambridge, pages 223–231.

Matthew Snover, Nitin Madnani, Bonnie J.Dorr and Richard Schwartz. 2009. Fluency, adequacy, or HTER? Exploring different human judgments with a tunable MT metric. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, Athens, Greece, pages 259–268.

Ying Zhang and Stephan Vogel. 2004. Measuring Confidence Intervals for the Machine Translation Evaluation Metrics. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI)*, pages 85–94.

Andreas Zollmann and Ashish Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proceedings of HLT-NAACL 2006: Proceedings of the Workshop on Statistical Machine Translation*, New York, pages 138–141.