Improving Statistical Machine Translation Accuracy Using Bilingual Lexicon Extraction with Paraphrases

Chenhui Chu[1,3], **Toshiaki Nakazawa**[2], **Sadao Kurohashi**[1]
[1]Graduate School of Informatics, Kyoto University
[2]Japan Science and Technology Agency
[3]Japan Society for the Promotion of Science Research Fellow
chu@nlp.ist.i.kyoto-u.ac.jp, nakazawa@pa.jst.jp, kuro@i.kyoto-u.ac.jp

Abstract

Statistical machine translation (SMT) suffers from the *accuracy problem* that the translation pairs and their feature scores in the translation model can be inaccurate. The *accuracy problem* is caused by the quality of the unsupervised methods used for translation model learning. Previous studies propose estimating comparable features for the translation pairs in the translation model from comparable corpora, to improve the accuracy of the translation model. Comparable feature estimation is based on bilingual lexicon extraction (BLE) technology. However, BLE suffers from the data sparseness problem, which makes the comparable features inaccurate. In this paper, we propose using paraphrases to address this problem. Paraphrases are used to smooth the vectors used in comparable feature estimation with BLE. In this way, we improve the quality of comparable features, which can improve the accuracy of the translation model thus improve SMT performance. Experiments conducted on Chinese-English phrase-based SMT (PBSMT) verify the effectiveness of our proposed method.

1 Introduction

In statistical machine translation (SMT) (Brown et al., 1993), the translation model is automatically learned form parallel corpora in an unsupervised way. The translation model contains translation pairs with their features scores. SMT suffers from the *accuracy problem* that the translation model may be inaccurate, meaning that the translation pairs and their features scores may be inaccurate. The *accuracy problem* is caused by the quality of the unsupervised method used for translation model learning, which always correlates with the amount of parallel corpora. Increasing the amount of parallel corpora is a possible way to improve the accuracy, however parallel corpora remain a scarce resource for most language pairs and domains.[1] Accuracy also can be improved by filtering out the noisy translation pairs from the translation model, however meanwhile we may lose some good translation pairs, thus the coverage of the translation model may decrease. A good solution to improve the accuracy while keeping the coverage is estimating new features for the translation pairs from comparable corpora (which we call comparable features), to make the translation model more discriminative thus more accurate.

Previous studies use bilingual lexicon extraction (BLE) technology to estimate comparable features (Klementiev et al., 2012; Irvine and Callison-Burch, 2013a). They extend traditional BLE that estimates similarity for bilingual word pairs on comparable corpora, to translation pairs in the translation model of SMT. The similarity scores of the translation pairs are used as comparable features. These comparable features are combined with the original features used in SMT, which can provide additional information to distinguish good and bad translation pairs. A major problem of previous studies is that they do not deal with the data sparseness problem that BLE suffers from. BLE uses vector representations for word

[1] Scarceness of parallel corpora also leads to the low coverage of the translation model (which we call the *coverage problem* of SMT), however we do not tackle this in this paper.

pairs to compare the similarity between them. Data sparseness makes the vector representations sparse (e.g., the vector of a low frequent word tends to have many zero entries), thus they do not always reliably represent the meanings of words. Therefore, the similarity of word pairs can be inaccurate. Smoothing technology has been proposed to address the data sparseness problem for BLE. Pekar et al. (2006) smooth the vectors of words with their distributional nearest neighbors, however distributional nearest neighbors can have different meanings and thus introduce noise. Andrade et al. (2013) use synonym sets in WordNet to smooth the vectors of words, however WordNet is not available for every language. More importantly, both studies work for words, which are not suitable for comparable feature estimation. The reason is that translation pairs can also be phrases (Koehn et al., 2003) or syntactic rules (Galley et al., 2004) etc., depending on what kind of SMT models we use.

In this paper, we propose using paraphrases to address the data sparseness problem of BLE for comparable feature estimation. A paraphrase is a restatement of the meaning of a word, phrase or syntactic rule etc., therefore it is suitable for the data sparseness problem. We generate paraphrases from the parallel corpus used for translation model learning. Then, we use the paraphrases to smooth the vectors of the translation pairs in the translation model for comparable feature estimation. Smoothing is done by learning vectors that combine the vectors of the original translation pairs with the vectors of their paraphrases. The smoothed vectors can overcome the data sparseness problem, making the vectors more accurately represent the meanings of the translation pairs. In this way, we improve the quality of comparable features, which can improve the accuracy of the translation model thus improve SMT performance.

We conduct experiments on Chinese-English Phrase-based SMT (PBSMT) (Koehn et al., 2003).[2] Experimental results show that our proposed method can improve SMT performance, compared to the previous studies that estimate comparable features without dealing with the data sparseness problem of BLE (Klementiev et al., 2012; Irvine and Callison-Burch, 2013a). The results verify the effectiveness of using BLE together with paraphrases for the *accuracy problem* of SMT.

2 Related Work

2.1 Bilingual Lexicon Extraction (BLE) for SMT

From the pioneering work of (Rapp, 1995), BLE from comparable corpora has been studied for a long time. BLE is based on the distributional hypothesis (Harris, 1954), stating that words with similar meaning have similar distributions across languages. Contextual similarity (Rapp, 1995), topical similarity (Vulić et al., 2011) and temporal similarity (Klementiev and Roth, 2006) can be important clues for BLE. Orthographic similarity may also be used for BLE for some similar language pairs (Koehn and Knight, 2002). Moreover, some studies try to use the combinations of different similarities for BLE (Irvine and Callison-Burch, 2013b; Chu et al., 2014). To address the data sparseness problem of BLE, smoothing technology has been proposed (Pekar et al., 2006; Andrade et al., 2013).

BLE can be used to address the *accuracy problem* of SMT, which estimates comparable features for the translation pairs in the translation model (Klementiev et al., 2012). BLE also can be used to address the *coverage problem* of SMT, which mines translations for the unknown words or phrases in the translation model from comparable corpora (Daume III and Jagarlamudi, 2011; Irvine et al., 2013). Moreover, studies have been conducted to address the *accuracy and coverage problems* of SMT simultaneously with BLE (Irvine and Callison-Burch, 2013a).

Our study focuses on addressing the *accuracy problem* of SMT with BLE. We use paraphrases to address the data sparseness problem of BLE for comparable feature estimation, which makes the comparable features more accurate.

2.2 Paraphrases for SMT

Many methods have been proposed to use paraphrases for SMT, mainly for the *coverage problem.* One method is paraphrasing unknown words or phrases in the translation model (Callison-Burch et al., 2006; Razmara et al., 2013; Marton et al., 2009).

[2] Our proposed method can also be applied to other language pairs and SMT models.

f	e	$\phi(f\|e)$	$lex(f\|e)$	$\phi(e\|f)$	$lex(e\|f)$	Alignment
失业 人数	**unemployment figures**	0.3	0.0037	0.0769	0.0018	0-0 1-1
失业 人数	**number of unemployed**	0.1333	0.0188	0.1025	0.0041	1-0 1-1 0-2
失业 人数	. unemployment was	0.3333	0.0015	0.0256	6.8e-06	0-1 1-1 1-2
失业 人数	unemployment and bringing	1	0.0029	0.0256	5.4e-07	0-0 1-0

Table 1: An example of the *accuracy problem* in PBSMT. The correct translations of "失业 (unemployment) 人数 (number of people)" are in bold. The incorrect phrase pairs are extracted because "人数 (number of people)" is incorrectly aligned to "unemployment", and their feature scores are incorrect.

Another method is constructing a paraphrase lattice for the tuning and testing data, and performing lattice decoding (Du et al., 2010; Bar and Dershowitz, 2014). Paraphrases also can be incorporated as additional training data, which may improve both coverage and accuracy of SMT (Pal et al., 2014).

Previous studies require external data in addition to the parallel corpus used for SMT for paraphrase generation to make their methods effective. These paraphrases can be generated from external parallel corpora (Callison-Burch et al., 2006; Du et al., 2010), or monolingual corpora based on distributional similarity (Marton et al., 2009; Razmara et al., 2013; Pal et al., 2014; Bar and Dershowitz, 2014).

Our study differs from previous studies in using paraphrases for smoothing the vectors of BLE, which is used for comparable feature estimation that can improve the accuracy of SMT. Another difference is that our proposed method is effective when only using the paraphrases generated from the parallel corpus used for SMT, while previous studies require external data for paraphrase generation.

3 Accuracy Problem of Phrase-based SMT (PBSMT)

In this paper, we conduct experiments on PBSMT (Koehn et al., 2003). Here, we give a brief overview of PBSMT, and explain the *accuracy problem* of PBSMT.

In PBSMT, the translation model is represented as a phrase table, containing phrase pairs together with their feature scores.[3] The phrase pairs are extracted based on unsupervised word alignments, whose quality always correlates with the amount of the parallel corpus. Inverse and direct phrase translation probabilities $\phi(f|e)$ and $\phi(e|f)$, inverse and direct lexical weighting $lex(f|e)$ and $lex(e|f)$ are used as features for the phrase table. Phrase translation probabilities are calculated via maximum likelihood estimation, which counts how often a source phrase f is aligned to target phrase e in the parallel corpus, and vise versa. Lexical weighting is the average word translation probability calculated using internal word alignments of a phrase pair, which is used to smooth the overestimation of the phrase translation probabilities. Other typical features such as the reordering model features and the n-gram language model features are also used in PBSMT. These features are combined in a log linear model, and their weights are tuned using a small size of parallel sentences. During decoding, these features together with their tuned weights are used to produce new translations.

One problem of PBSMT is that the phrase pairs and their feature scores in the phrase table may be inaccurate. One reason for this is the quality of the word alignment. Another reason is that the translation probabilities of rare word and phrase pairs tend to be grossly overestimated. Sparseness of the parallel corpus leads to word alignment errors and overestimations, which result in inaccurate phrase pairs and feature scores. Table 1 shows an example of phrase pairs and feature scores taken from the phrase table constructed in our experiments (See Section 5 for the details of the experiments), which contains inaccurate phrase pairs.

4 Proposed Method

Figure 1 shows an overview of our proposed method. We construct a phrase table from a parallel corpus following (Koehn et al., 2003). Because this phrase table may be inaccurate, we estimate comparable features from comparable corpora following (Klementiev et al., 2012; Irvine and Callison-Burch, 2013a). These comparable features are appended to the original phrase table, to address the *accuracy*

[3] Note that in PBSMT, the definition of a phrase also includes a single word.

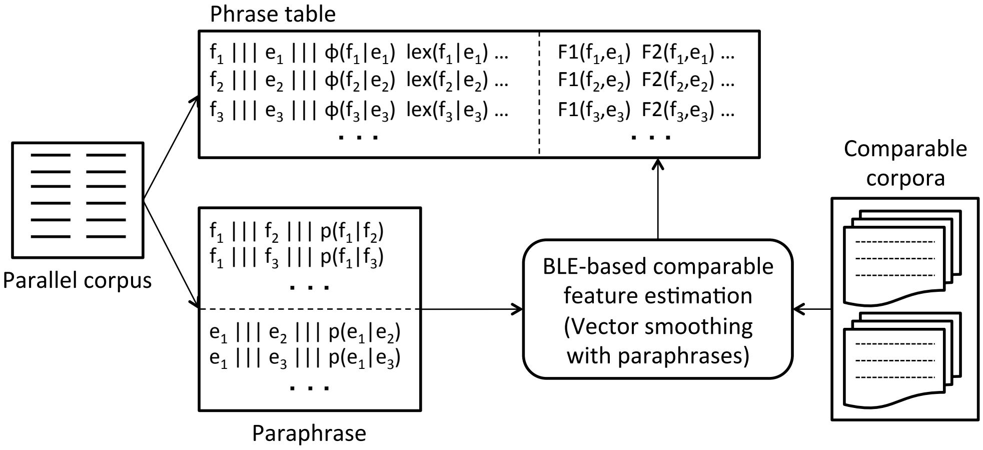

Figure 1: Overview of our proposed method.

problem of PBSMT. Comparable feature estimation is based on BLE, which suffers from the data sparseness problem. We propose using paraphrases to address this problem. We generate phrasal level paraphrases for both the source and target language from the parallel corpus. Then we use the paraphrases to smooth the vectors of the source and target phrases used for comparable feature estimation respectively. Smoothing is done by learning a vector that combines the original vector of a phrase with the vectors of its paraphrases. The smoothed vectors can represent the meanings of phrase pairs more accurately. Finally, we compute the similarity of phrase pairs based on the smoothed source and target vectors. In this way, we improve the quality of comparable features, which can improve the accuracy of the phrase table thus improve SMT performance.

Details of paraphrase generation, comparable feature estimation and vector smoothing with paraphrases will be described in Section 4.1, 4.2 and 4.3 respectively.

4.1 Paraphrase Generation

In this paper, we generate both source and target phrasal level paraphrases from the parallel corpus used for SMT[4] through bilingual pivoting (Bannard and Callison-Burch, 2005). The idea of this method is that if two source phrases f_1 and f_2 are translated to the same target phrase e, we can assume that f_1 and f_2 are a paraphrase pair. Probability of this paraphrase pair can be assigned by marginalizing over all shared target translations e in the parallel corpus, defined as follows:

$$p(f_1|f_2) = \sum_{e} \phi(f_1|e)\phi(e|f_2) \qquad (1)$$

where, $\phi(f_1|e)$ and $\phi(e|f_2)$ are phrase translation probability. Target paraphrases can be generated in a similar way.

Note that word alignment errors can also lead to incorrect paraphrase generation. For example, “unemployment figures” and “unemployment and bringing” in Table 1 might be generated as a paraphrase pair. However, this kind of noisy pairs can be easily pruned according to their low probabilities.

4.2 Comparable Feature Estimation

Following (Klementiev et al., 2012; Irvine and Callison-Burch, 2013a), we estimate contextual, topical and temporal similarities as comparable features. However, we do not use orthographic similarity as comparable feature, because we experiment on Chinese-English, which is not an orthographically similar language pair.

Besides phrasal features, we also estimate lexical features following (Klementiev et al., 2012; Irvine and Callison-Burch, 2013a). The lexical features are the average similarity scores of word pairs over all possible word alignments across two phrases. They are used to smooth the phrasal features, like the lexical weighting in PBSMT. However, they only can slightly alleviate the sparseness of phrasal features, because individual words also suffer from the data sparseness problem.

In the following sections, we describe the meth-

[4] Paraphrases also can be generated from external parallel corpora and monolingual corpora, however we leave it as future work.

ods to estimate contextual, topical and temporal features in detail.

Contextual feature

Contextual feature is the contextual similarity of a phrase pair. Contextual similarity is based on the distributional hypothesis on context, stating that phrases with similar meaning appear in similar contexts across languages. From the pioneering work of (Rapp, 1995), contextual similarity has been used for BLE for a long time.

In the literature, different definitions of context have been proposed for BLE, such as window-based context, sentence-based context and syntax-based context etc. In this paper, we use window-based context, and leave the comparison of using different definitions of context as future work. Given a phrase, we count all its immediate context words, with a window size of 4 (2 preceding words and 2 following words). We build a context by collecting the counts in a bag of words fashion, namely we do not distinguish the positions that the context words appear in. The number of dimensions of the constructed vector is equal to the vocabulary size. We further reweight each component in the vector by multiplying by the *IDF* score following (Garera et al., 2009; Chu et al., 2014), which is defined as follows:

$$IDF(t, D) = log\frac{|D|}{1 + |\{d \in D : t \in d\}|} \quad (2)$$

where $|D|$ is the total number of documents in the corpus, and $|\{d \in D : t \in d\}|$ denotes number of documents where the term t appears.[5] We model the source and target vectors using the method described above, and project the source vector onto the vector space of the target language using a seed dictionary. The contextual similarity of the phrase pair is the similarity of the vectors, which is computed using cosine similarity defined as follows:

$$Cos(f, e) = \frac{\sum_{k=1}^{K} F_k \times E_k}{\sqrt{\sum_{k=1}^{K}(F_k)^2} \times \sqrt{\sum_{k=1}^{K}(E_k)^2}} \quad (3)$$

where f and e are the source and target phrases, F and E are the projected source vector and target vector, K is the number of dimensions of the vectors.

Topical feature

Topical feature is the topical similarity of a phrase pair. Topical similarity uses the distributional hypothesis on topics, stating that two phrases are potential translation candidates if they are often present in the same cross-lingual topics and not observed in other cross-lingual topics (Vulić et al., 2011). Vulić et al. (2011) propose using bilingual topic model based method to estimate topical similarity. However, this method is not scalable for large data sets.

In this paper, we estimate topical feature in a scalable way following (Klementiev et al., 2012). We treat an article pair aligned by interlanguage links in Wikipedia as a topic aligned pair. For a phrase pair, we build source and target topical occurrence vectors by counting their occurrences in its corresponding language articles. The number of dimensions of the constructed vector is equal to the number of aligned article pairs, and each dimension is the number of times that the phrase appears in the corresponding article. The similarity of the phrase pair is computed as the similarity of the source and target vectors using cosine similarity (Equation 3).

Temporal feature

Temporal feature is the temporal similarity of a phrase pair. The intuition of temporal similarity is that news stories across languages tend to discuss the same world events on the same day, and the occurrences of a translated phrase pair over time tend to spike on the same dates (Klementiev and Roth, 2006; Klementiev et al., 2012).

We estimate temporal feature following (Klementiev and Roth, 2006; Klementiev et al., 2012). For a phrase pair, we build source and target temporal occurrence vectors by counting their occurrences in equally sized temporal bins, which are sorted from the set of time-stamped documents in the comparable corpus. We set the window size of a bin to 1 day. Therefore the number of dimensions of the constructed vector is equal to the number of days spanned by the corpus, and each dimension is the number of times that the phrase appears in the corresponding bin. The similarity of the phrase pair is computed as the similarity of the source and target vectors using cosine similarity (Equation 3).

4.3 Vector Smoothing with Paraphrases

Data sparseness results in sparse representations of the vectors, therefore the similarity of the phrase pair can be inaccurate. We propose using paraphrases to

[5] Since there are no document bounds in the corpus we used to estimate contextual feature, we treated every 100 sentences as one document.

Phrase	Paraphrase
tampered	being tampered
an appropriation	appropriation
11th	11th .
so many years	many years
first thing	first thing that
mass media ,	media ,

Table 2: Examples of overlaps between a phrase and its paraphrase.

smooth both the source and target vectors, to deal with the data sparseness problem. After smoothing, the vectors can more accurately represent the phrases. We compute the similarity of the phrase pair based on the smoothed source and target vectors, and use it as comparable features for PBSMT.

One problem of using paraphrases for smoothing is that a phrase and its paraphrase may overlap. Table 2 shows some examples of overlaps between a phrase and its paraphrase generated from the parallel corpus we use. The vector of the overlapped paraphrase contains overlapped information of the vector of the original phrase. Therefore, it is necessary to consider overlap when using paraphrases for vector smoothing.

There are three types of vectors (context, topical and temporal occurrence vectors) need to be smoothed. The method for smoothing context vector is different from topical and temporal occurrence vectors, because the components in context vector are different. Topical and temporal occurrence vectors can be smoothed using the same method, because the components of both vectors are occurrence information. The following sections describe the methods to smooth the context vector, and topical and temporal occurrence vectors respectively.

Context Vector Smoothing

We smooth the context vector of a phrase x with the following equation:

$$X' = \frac{f(x)}{f(x) + \sum_{j=1}^{n} f(x_j)} \cdot X + \sum_{i=1}^{n} \frac{f(x_i)}{f(x) + \sum_{j=1}^{n} f(x_j)} \cdot p(x_i|x) \cdot \begin{cases} X_i \backslash X & (x \subset x_i) \\ X_i - X & (x \supset x_i) \\ X_i & (otherwise) \end{cases} \quad (4)$$

where X' is the smoothed context vector, X is the context vector of x, n is the number of paraphrases that x has, X_i is the context vector of paraphrase x_i, $p(x_i|x)$ is the probability that x_i is a paraphrase of x. $f(x)$ is the frequency of x in the corpus, and $\frac{f(x)}{f(x)+\sum_{j=1}^{n} f(x_j)}$ is the frequency weight for x. Frequency weight is also used for the paraphrases in a similar way. The frequency weight is proposed by Andrade et al. (2013) when using synonyms to smooth the context vector of a word. They show that using the frequency information of words as weights performs better than simple summation of the vectors. For the overlap problem between x and x_i, we do the following:

- If $x \subset x_i$ namely x is contained in x_i, we use the context words that exist in X_i but do not exist in X for smoothing, which is $X_i \backslash X$;
- If $x \supset x_i$ namely x contains x_i, we remove the overlapped contextual information between X_i and X for smoothing, which is $X_i - X$;
- Otherwise, we use X_i for smoothing.

Topical and Temporal Occurrence Vectors Smoothing

We smooth the topical and temporal occurrence vectors of a phrase x with the following equation:

$$X' = X + \sum_{i=1}^{n} p(x_i|x) \cdot \begin{cases} 0 & (x \subset x_i) \\ X_i - X & (x \supset x_i) \\ X_i & (otherwise) \end{cases} \quad (5)$$

where X' is the smoothed occurrence vector, X is the occurrence vector of x, n is the number of paraphrases that x has, X_i is the occurrence vector of paraphrase x_i, $p(x_i|x)$ is the probability that x_i is a paraphrase of x. For the overlap problem between x and x_i, we do the following:

- If $x \subset x_i$, we do not use X_i for smoothing, because X already contains the occurrence information in X_i;
- If $x \supset x_i$, we remove the overlapped occurrence information between X_i and X for smoothing, which is $X_i - X$;
- Otherwise, we use X_i for smoothing.

Examples of the three types of vectors before and after smoothing are shown in Table 3.

	Before smoothing	After smoothing
Context	<rising: 2.37, economic: 0, recession: 3.94 $\cdots$ >	<rising: 0.03, economic: 0.06, recession: 0.04 $\cdots$ >
Topical	<Topic1: 0, Topic2: 1, Topic3: 0 $\cdots$ >	<Topic1: 0.12, Topic2: 1.27, Topic3: 0.05 $\cdots$ >
Temporal	<Date1: 1, Date2: 0, Date3: 6 $\cdots$ >	<Date1: 1.25, Date2: 0.08, Date3: 6.38 $\cdots$ >

Table 3: Examples of the three types of vectors for the phrase "unemployment figures" before and after smoothing.

5 Experiments

In our experiments, we compared our proposed method with (Klementiev et al., 2012). We estimated comparable features from comparable corpora using the method of (Klementiev et al., 2012) and our proposed method respectively. We appended the comparable features to the phrase table, and evaluated the two methods in the perspective of SMT performance. We conducted experiments on Chinese-English data. In all our experiments, we preprocessed the data by segmenting Chinese sentences using a segmenter proposed by Chu et al. (2012), and tokenizing English sentences.

5.1 Experimental Settings

SMT Settings

We conducted Chinese-to-English translation experiments. The parallel corpus we used is from Chinese-English NIST open MT.[6] The "NIST" column of Table 4 shows the statistics of this parallel corpus. For decoding, we used the state-of-the-art PBSMT toolkit Moses (Koehn et al., 2007) with default options, except for the phrase length limit (7→3) following (Klementiev et al., 2012). We trained a 5-gram language model on the English side of the parallel corpus using the SRILM toolkit[7] with interpolated Kneser-Ney discounting, and used it for all the experiments. We used NIST open MT 2002 and 2003 data sets for tuning and testing, containing 878 and 919 sentence pairs respectively. Note that both MT 2002 and 2003 data sets contain 4 references for each Chinese sentence. Tuning was performed by minimum error rate training (MERT) (Och, 2003), and it was re-run for every experiment.

Comparable Feature Estimation Settings

	NIST	Gigaword	Wikipedia
# Zh articles	N/A	3.6M	248k
# En articles	N/A	4.3M	248k
# Zh sentences	991k	42.6M	2.8M
# En sentences	991k	56.9M	10.1M
# Zh tokens	26.1M	1.1B	70.5M
# En tokens	27.2M	1.3B	240.5M

Table 4: Statistics of the comparable data used for comparable feature estimation.

Table 4 shows the statistics of the comparable data used for comparable feature estimation. The contextual feature was estimated on the parallel corpus. We treated the two sides of the parallel corpus as independent monolingual corpora, following (Haghighi et al., 2008; Klementiev et al., 2012). Contextual feature estimation requires a seed dictionary. The seed dictionary we used is NIST Chinese-English translation lexicon Version 3.0,[8] containing 82k entries. The temporal feature was estimated on Chinese[9] and English[10] Gigaword version 5.0. We used the afp, cna and xin sections with date range 1994/05-2010/12 of the corpora. The topical feature was estimated on Chinese and English Wikipedia data. We downloaded Chinese[11] (2012/09/21) and English[12] (2012/10/01) Wikipedia database dumps. We used an open-source Python script[13] to extract and clean the text from the dumps. We aligned the articles on the same topic in Chinese-English Wikipedia via the interlanguage links.

We estimated comparable features for the unique phrase pairs used for tuning and testing. These phrase pairs were extracted from the entire phrase table constructed from the parallel corpus, by checking all the source phrases in the tuning and testing data sets. We call these phrase pairs the filtered phrase table. Table 5 shows the statistics of the fil-

[6] LDC2007T02, LDC2002T01, LDC2003T17, LDC2004T07, HK News part of LDC2004T08, LDC2005T10 and LDC2006T04

[7] http://www.speech.sri.com/projects/srilm

[8] LDC2002L27

[9] LDC2011T13

[10] LDC2011T07

[11] http://dumps.wikimedia.org/zhwiki

[12] http://dumps.wikimedia.org/enwiki

[13] http://code.google.com/p/recommend-2011/source/browse/Ass4/WikiExtractor.py

# Phrase pairs	4,886,067
# Zh phrases	45,905
# En phrases	2,078,230
# Zh unigrams	6,719
Avg # translations	509.1
# Zh bigrams	23,029
Avg # translations	56.7
# Zh trigrams	16,157
Avg # translations	9.8

Table 5: Statistics of the filtered phrase table.

	Zh	En
# Phrases&words	46,112	2,090,345
# Phrases&words w/ paraphrases	26,718	455,099
# Unigrams w/ paraphrases	6,273	46,191
# paraphrases	39.8	21.6
# Bigrams w/ paraphrases	15,026	223,299
Avg # paraphrases	34.6	17.7
# Trigrams w/ paraphrases	5,419	185,609
# paraphrases	20.0	14.9

Table 6: Statistics the generated paraphrases for the phrases and individual words inside the phrases in the filtered phrase table.

tered phrase table. We can see that each Chinese phrase has a large number of translations on average especially for the lower order n-gram phrases, which can indicate the inaccuracy of the filtered phrase table.

Our proposed method requires paraphrases for vector smoothing. We used Joshua (Ganitkevitch et al., 2012) to generate both Chinese and English paraphrases from the parallel corpus. We kept the paraphrase pairs that satisfy $logp(x_1|x_2) > -7$ and $logp(x_2|x_1) > -7$ [14] for smoothing, where $p(x_1|x_2)$ is the probability that x_1 is a paraphrase of x_2, and $p(x_2|x_1)$ is the probability that x_2 is a paraphrase of x_1. Table 6 shows the statistics of the paraphrase generation results for the Chinese and English phrases, and individual words inside the phrases in the filtered phrase table.

Note that, for some phrase pairs, their comparable feature scores may be 0, because of data sparseness. In that case, we set their comparable features to a small positive number of $1e - 07$.

System	+Contextual	+Topical	+Temporal	+All
Baseline	45.45			
Klementiev+	43.69	45.72	45.05	45.92
Proposed	45.56‡	46.10†‡	46.00†‡	**46.26**†

Table 7: BLEU-4 scores for Chinese-to-English translation experiments ("†" and "‡" denote that the result is significantly better than "Baseline" at $p < 0.01$ and "Klementiev+" at $p < 0.05$ respectively)

5.2 Results

We report results on the test set using case-insensitive BLEU-4 score and four references. Table 7 shows the results of Chinese-to-English translation experiments. "Baseline" denotes the baseline system that does not use comparable features. "Klementiev+" denotes the system that appends the comparable features estimated following (Klementiev et al., 2012) to the phrase table. "Proposed" denotes the system that uses the comparable features estimated by our proposed method. "+Contextual", "+Topical" and "+Temporal" denote the systems that append contextual, topical and temporal features respectively. "+All" denotes the system that appends all the three types of features. The significance test was performed using the bootstrap resampling method proposed by Koehn (2004).

We can see that "Klementiev+" does not always outperform "Baseline". The reason for this is that the comparable features estimated by (Klementiev et al., 2012) are inaccurate. "Proposed" performs significantly better than both "Baseline" and "Klementiev+". The reason for this is that "Proposed" deals with the data sparseness problem of BLE for comparable feature estimation, making the features more accurate thus improve the SMT performance. As for different comparable features of "Proposed", "+Contextual", "+Topical" and "+Temporal" are all helpful, and combining them can be more effective. The results verify the effectiveness of our proposed method for the *accuracy problem* of PBSMT.

We also investigated the comparable features estimated by the method of (Klementiev et al., 2012) and our proposed method. Based on our investigation, most comparable features estimated by our proposed method are more accurate than the ones estimated by the method of (Klementiev et al., 2012). Here, we give an example of the comparable fea-

[14] We also tried other pruning thresholds, and this threshold showed the best performance in the preliminary experiments.

f	e	con	con_lex	top	top_lex	tem	tem_lex
失业 人数	**unemployment figures**	1.4e-06	0.0408	1e-07	0.2061	0.1942	0.6832
失业 人数	**number of unemployed**	0.0144	0.0299	1e-07	0.1675	0.0236	0.6277
失业 人数	. unemployment was	0.0107	0.0701	1e-07	0.1908	0.0709	0.6981
失业 人数	unemployment and bringing	1e-07	0.0603	1e-07	0.1730	1e-07	0.6898
失业 人数	**unemployment figures**	0.0749	0.0806	0.5434	0.2629	0.4307	0.7033
失业 人数	**number of unemployed**	0.0522	0.1053	0.1907	0.2235	0.5983	0.7240
失业 人数	. unemployment was	0.0050	0.1206	0.0117	0.2336	0.0967	0.7094
失业 人数	unemployment and bringing	5.1e-05	0.0904	1e-07	0.2034	0.0073	0.7003

Table 8: Examples of comparable feature scores estimated by the method of (Klementiev et al., 2012) (above the bold line) and our proposed method (below the bold line) for the phrase pairs shown in Table 1 ("con", "top" and "tem" denote phrasal contextual, topical and temporal features respectively, "con_lex", "top_lex" and "tem_lex" denote lexical contextual, topical and temporal features respectively).

ture scores estimated for the phrase pairs shown in Table 1. Table 8 shows the comparable feature scores estimated by the method of (Klementiev et al., 2012) (above the bold line) and our proposed method (below the bold line). We can see that the method of (Klementiev et al., 2012) suffers from the data sparseness problem. Many of the feature scores are $1e - 07$, and many of the feature scores for the correct translations ("unemployment figures" and "number of unemployed") are lower than the incorrect ones (". unemployment was" and "unemployment and bringing"). Our proposed method addresses the data sparseness problem by using paraphrases for vector smoothing. We can see that, after smoothing the feature scores can more accurately distinguish the good translations from the bad ones.

6 Conclusion and Future Work

In this paper, we proposed using BLE together with paraphrases to address the *accuracy problem* of SMT. The translation pairs and their feature scores in the translation model of SMT can be inaccurate, because of the quality of the unsupervised methods used for translation model learning. Estimating comparable features from comparable corpora with BLE has been proposed for the *accuracy problem* of SMT. However, BLE suffers from the data sparseness problem, which makes the comparable features inaccurate. We proposed using paraphrases to address this problem. Paraphrases were used to smooth the vectors used in comparable feature estimation with BLE. Experiments conducted on Chinese-English PBSMT verified the effectiveness of our proposed method.

As future work, firstly we plan to generate paraphrases from external parallel corpora and monolingual corpora, where as in this paper we used the paraphrases generated from the parallel corpus used for SMT. Secondly, in this paper we estimated contextual features from the parallel corpus, however in the future we plan to estimate it from comparable corpora. Finally, since our proposed method should be language independent and can be applied to other SMT models, we plan to conduct experiments on other language pairs and SMT models to verify this.

Acknowledgement

This work was supported by the Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for JSPS Fellows. We also thank the anonymous reviewers for their valuable comments.

References

Daniel Andrade, Masaki Tsuchida, Takashi Onishi, and Kai Ishikawa. 2013. Translation acquisition using synonym sets. In *Proceedings of NAACL-HLT 2013*, pages 655–660.

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of ACL 2005*, pages 597–604.

Kfir Bar and Nachum Dershowitz. 2014. Inferring paraphrases for a highly inflected language from a monolingual corpus. In *Proceedings of CICLing 2014*, pages 8404:2:245–256.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Association for Computational Linguistics*, 19(2):263–312.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In *Proceedings of NAACL-HLT 2006*, pages 17–24.

Chenhui Chu, Toshiaki Nakazawa, Daisuke Kawahara, and Sadao Kurohashi. 2012. Exploiting shared Chinese characters in Chinese word segmentation optimization for Chinese-Japanese machine translation. In *Proceedings of EAMT 2012*, pages 35–42.

Chenhui Chu, Toshiaki Nakazawa, and Sadao Kurohashi. 2014. Iterative bilingual lexicon extraction from comparable corpora with topical and contextual knowledge. In *Proceedings of CICLing 2014*, pages 8404:2:296–309.

Hal Daume III and Jagadeesh Jagarlamudi. 2011. Domain adaptation for machine translation by mining unseen words. In *Proceedings of ACL-HLT 2011*, pages 407–412.

Jinhua Du, Jie Jiang, and Andy Way. 2010. Facilitating translation using source language paraphrase lattices. In *Proceedings of EMNLP 2010*, pages 420–429.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In Daniel Marcu Susan Dumais and Salim Roukos, editors, *Proceedings of NAACL-HLT 2004*, pages 273–280.

Juri Ganitkevitch, Yuan Cao, Jonathan Weese, Matt Post, and Chris Callison-Burch. 2012. Joshua 4.0: Packing, pro, and paraphrases. In *Proceedings of WMT 2012*, pages 283–291.

Nikesh Garera, Chris Callison-Burch, and David Yarowsky. 2009. Improving translation lexicon induction from monolingual corpora via dependency contexts and part-of-speech equivalences. In *Proceedings of CoNLL 2009*, pages 129–137.

Aria Haghighi, Percy Liang, Taylor Berg-Kirkpatrick, and Dan Klein. 2008. Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-HLT 2008*, pages 771–779.

Zellig S. Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Ann Irvine and Chris Callison-Burch. 2013a. Combining bilingual and comparable corpora for low resource machine translation. In *Proceedings of WMT 2013*, pages 262–270.

Ann Irvine and Chris Callison-Burch. 2013b. Supervised bilingual lexicon induction with multiple monolingual signals. In *Proceedings of NAACL-HLT 2013*, pages 518–523.

Ann Irvine, Chris Quirk, and Hal Daumé III. 2013. Monolingual marginal matching for translation model adaptation. In *Proceedings of EMNLP 2013*, pages 1077–1088.

Alexandre Klementiev and Dan Roth. 2006. Weakly supervised named entity transliteration and discovery from multilingual comparable corpora. In *Proceedings of COLING-ACL 2006*, pages 817–824.

Alexandre Klementiev, Ann Irvine, Chris Callison-Burch, and David Yarowsky. 2012. Toward statistical machine translation without parallel corpora. In *Proceedings of EACL 2012*, pages 130–140.

Philipp Koehn and Kevin Knight. 2002. Learning a translation lexicon from monolingual corpora. In *Proceedings of the ACL-02 Workshop on Unsupervised Lexical Acquisition*, pages 9–16.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL-HLT 2003*, NAACL '03, pages 48–54.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL 2007*, pages 177–180.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 388–395.

Yuval Marton, Chris Callison-Burch, and Philip Resnik. 2009. Improved statistical machine translation using monolingually-derived paraphrases. In *Proceedings of EMNLP 2009*, pages 381–390.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL 2003*, pages 160–167.

Santanu Pal, Pintu Lohar, and Sudip Kumar Naskar. 2014. Role of paraphrases in pb-smt. In *Proceedings of CICLing 2014*, pages 8404:2:245–256.

Viktor Pekar, Ruslan Mitkov, Dimitar Blagoev, and Andrea Mulloni. 2006. Finding translations for low-frequency words in comparable corpora. *Machine Translation*, 20(4):247–266.

Reinhard Rapp. 1995. Identifying word translations in non-parallel texts. In *Proceedings of ACL 1995*, pages 320–322.

Majid Razmara, Maryam Siahbani, Reza Haffari, and Anoop Sarkar. 2013. Graph propagation for paraphrasing out-of-vocabulary words in statistical machine translation. In *Proceedings of ACL 2013*, pages 1105–1115.

Ivan Vulić, Wim De Smet, and Marie-Francine Moens. 2011. Identifying word translations from comparable corpora using latent topic models. In *Proceedings of ACL-HLT 2011*, pages 479–484.