Collective Corpus Weighting and Phrase Scoring for SMT Using Graph-Based Random Walk

Lei Cui[1,⋆], Dongdong Zhang[2], Shujie Liu[2], Mu Li[2], and Ming Zhou[2]

[1] School of Computer Science and Technology
Harbin Institute of Technology, Harbin, China
leicui@hit.edu.cn
[2] Microsoft Research Asia, Beijing, China
{dozhang,shujliu,muli,mingzhou}@microsoft.com

Abstract. Data quality is one of the key factors in Statistical Machine Translation (SMT). Previous research addressed the data quality problem in SMT by corpus weighting or phrase scoring, but these two types of methods were often investigated independently. To leverage the dependencies between them, we propose an intuitive approach to improve translation modeling by collective corpus weighting and phrase scoring. The method uses the mutual reinforcement between the sentence pairs and the extracted phrase pairs, based on the observation that better sentence pairs often lead to better phrase extraction and vice versa. An effective graph-based random walk is designed to estimate the quality of sentence pairs and phrase pairs simultaneously. Extensive experimental results show that our method improves performance significantly and consistently in several Chinese-to-English translation tasks.

Keywords: data quality, corpus weighting, phrase scoring, graph-based random walk.

1 Introduction

Statistical Machine Translation (SMT) depends largely on the performance of translation modeling. The training of a translation model usually starts from automatically word-aligned bilingual corpus, followed by the extraction and scoring of phrase pairs. In real-world SMT systems, bilingual data is often mined from the web, meaning the low-quality data is inevitable. The low-quality bilingual data degrades the quality of word alignment and leads to incorrect phrase pairs and defective phrase scoring, which hurts the translation performance of phrase-based SMT systems. Therefore, it is crucial to exploit data quality information to improve the translation modeling.

Previous research has addressed the data quality problem by corpus weighting or phrase scoring, but these two kinds of methods are often investigated independently. On the one hand, conventional corpus weighting methods estimate the quality of each sentence pair in the bilingual corpus individually, but

⋆ This work has been done while the first author was visiting Microsoft Research Asia.

G. Zhou et al. (Eds.): NLPCC 2013, CCIS 400, pp. 176–187, 2013.

neglect that similar sentence pairs are usually in similar quality. On the other hand, the translation probability of phrase pairs are often estimated based on the assumption that the sentence pairs are equally well. In real-world SMT, these assumptions may not hold due to the varying quality of the bilingual corpus. Therefore, the mutual reinforcement should be leveraged for translation modeling, which means the quality of the sentence pairs depend on the extracted phrase pairs and vice versa.

To this end, we propose an intuitive and effective approach to address this problem. Obviously, high-quality parallel data tends to produce better phrase pairs than low-quality data. Meanwhile, it is also observed that the phrase pairs that appear frequently in the bilingual corpus are more reliable than less frequent ones because they are more reusable, hence most good sentence pairs are prone to contain more frequent phrase pairs [1, 2]. This kind of mutual reinforcement fits well into the framework of graph-based random walk. When a phrase pair p is extracted from a sentence pair s, s is considered casting a vote for p. The higher the number of votes a phrase pair has, the more reliable of the phrase pair. Similarly, the quality of the sentence pair s is also determined by the number of votes casted by all the extracted phrase pairs from s.

In this paper, we have developed a PageRank-style random walk algorithm [3, 4, 5] to iteratively compute the importance score of each sentence pair and each phrase pair that indicates its quality: the higher the better. In SMT, these scores are integrated into the log-linear model to help translation generation in decoding. The importance scores of sentence pairs are used as fractional counts to re-calculate the phrase translation probabilities based on Maximum Likelihood Estimation (MLE). In addition, the important scores of phrase pairs are directly used as new features. We evaluate our method on the colloquial text (IWSLT test sets) and the formal text (NIST test sets). Extensive experiments show that our method improves the performance significantly and consistently in several Chinese-to-English translation tasks, with up to 1.9 BLEU points.

The rest of the paper is organized as follows: The proposed approach is explained in Section 2. Experimental results are presented in Section 3. Section 4 introduces some related work. Section 5 concludes the paper and suggests future research directions.

2 The Proposed Approach

2.1 Graph-Based Random Walk

Graph-based random walk is a general algorithm for approximating the importance of a vertex within the graph in a global view. In our method, the vertices denote the sentence pairs and phrase pairs. The importance of each vertex is propagated to other vertices along the edges. Depending on different scenarios, the graph can take directed or undirected, weighted or un-weighted forms. Starting from the initial scores assigned in the graph, the algorithm is applied to recursively compute the importance scores of vertices until it converges, or the difference between two consecutive iterations falls below a pre-defined threshold.

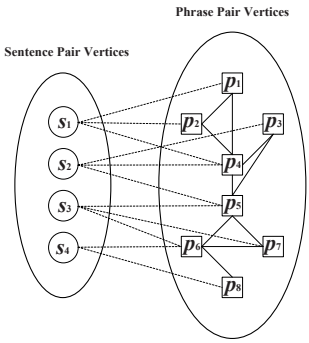

Fig. 1. The circular nodes stand for S and the square nodes stand for P. The dotted and solid lines capture the sentence-phrase and phrase-phrase recommendations.

2.2 Graph Construction

In this paper, we exploit the graph-based random walk to convert the translation modeling in SMT to a mutual recommendation problem. An undirected graph is constructed over the entire bilingual corpus. In the graph, we partition the vertices into two groups: sentence pair vertices and phrase pair vertices. The edges characterize the mutual recommendation relationships between vertices.

Formally, an undirected graph is defined as follows:

$$G = (V, E) \tag{1}$$

where $V = S \cup P$ is the vertex set, $E = E_{SP} \cup E_{PP}$ is the edge set. $S = \{s_i | 1 \leq i \leq n\}$ is the set of all sentence pairs. $P = \{p_j | 1 \leq j \leq m\}$ is the set of all phrase pairs that are extracted from S based on the word alignment. E_{SP} is a subset in which the edges are between S and P, thereby $E_{SP} = \{\langle s_i, p_j \rangle | s_i \in S, p_j \in P, \phi(s_i, p_j) = 1\}$.

$$\phi(s_i, p_j) = \begin{cases} 1 & \text{if } p_j \text{ can be extracted from } s_i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$E_{PP} = \{\langle p_j, p_k \rangle | p_j, p_k \in P, j \neq k, \psi(p_j, p_k) = 1\}$ denotes a subset of edges between vertices in P.

$$\psi(p_j, p_k) = \begin{cases} 1 & \exists i, \phi(s_i, p_j) = 1 \bigwedge \phi(s_i, p_k) = 1 \bigwedge A_j \bigcap A_k \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where A_j and A_k are the sets of word alignment links contained in phrase pairs p_j and p_k.

Figure 1 illustrates the graph structure. The circular nodes stand for S and the square nodes stand for P. The edges represent the mutual recommendation relationships among vertices. There are two types of recommendation relationships. The dotted lines capture the *Sentence-Phrase Recommendation* (SPR) denoted

by E_{SP} and the solid lines capture the *Phrase-Phrase Recommendation* (PPR) denoted by E_{PP}. If s_i is in high quality, it will recommend that p_j is a good phrase pair when $\phi(s_i, p_j) = 1$, and vice visa. Similarly, if p_j is a good phrase pair, it will recommend that the quality of p_k is good when $\psi(p_j, p_k) = 1$. The motivation behind this approach is that high quality sentence pairs can produce good phrase pairs. At the same time, a good phrase pair can recommend other phrase pairs that have good quality if they overlap within the same sentence pair. We do not consider the recommendation between sentence pairs because the dependency information between them is always missing during translation modeling in most SMT systems.

2.3 Graph Parameters

In general, graph parameters include importance scores for the vertices and weights for the edges. In our work, each edge is associated with a weight representing a recommendation score between two vertices. At the same time, each vertex is associated with a importance score representing the importance of the vertex within the graph.

The recommendation scores for SPR and PPR are computed in different ways. For SPR, a nonnegative recommendation score $h(s_i, p_j)$ is defined using the standard TF-IDF formula, which is similarly used in (Wan et al., 2007):

$$h(s_i, p_j) = \begin{cases} \frac{PF(s_i,p_j) \times IPF(p_j)}{\sum_{p' \in \{p|\phi(s_i,p)=1\}} PF(s_i,p') \times IPF(p')} & \text{if } \phi(s_i, p_j) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $PF(s_i, p_j)$ is the phrase frequency in a sentence pair and $IPF(p_j)$ is the inverse sentence frequency of p_j in the whole parallel corpus, $h(s_i, p_j)$ can be abbreviated as h_{ij} if there is no ambiguity.

For PPR, its recommendation score is defined based on the statistics of word alignment links. Given an edge $\langle p_j, p_k \rangle \in E_{PP}$, let A_m be the set of word alignment links contained in a phrase pair p_m, the recommendation score of $\langle p_j, p_k \rangle$ is computed by the Dice's coefficient:

$$g(p_j, p_k) = \frac{2|A_j \bigcap A_k|}{|A_j| + |A_k|} \tag{5}$$

The larger $g(p_j, p_k)$ is, the more contexts between p_j and p_k are shared and thereby the stronger the recommendation between them. $g(p_j, p_k)$ is abbreviated as g_{jk}. Figure 2 illustrates an example of PPR based on the word alignment generated by GIZA++. p_1 and p_2 have strong mutual recommendation as well as p_3 and p_4, because they share many word alignment links, while p_2 and p_3 have less mutual recommendation. This kind of mutual recommendation is propagated among the vertices along the edges in the graph. Furthermore, through human checking, we find that p_3 is not a good phrase pair due to the word alignment error. Naturally, this strongly suggests that p_4 is a poor phrase pair but weakly suggests that p_2 is poor. Finally, the quality of each phrase pair is determined by the net effect of the positive and negative recommendations.

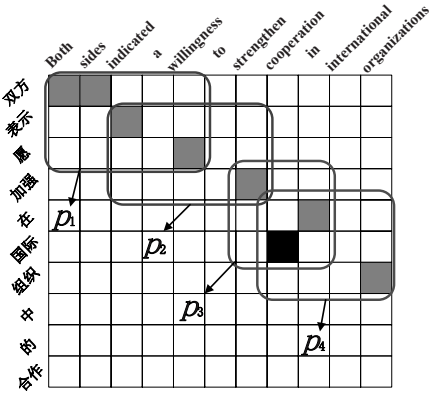

Fig. 2. A PPR example, $\langle guoji, cooperation \rangle$ is an incorrect link shared by p_3 and p_4

2.4 Weighted Mutual Recommendation Algorithm

In this section, we develop an algorithm to estimate the importance scores of the vertices using the weights of the edges in the graph. The importance score of a vertex is influenced by either the neighboring sentence pair vertices or the neighboring phrase pair vertices. Extending the PageRank algorithm [3], Mihalcea and Tarau [4] proposed a graph-based random walk algorithm for weighted graphs to calculate the importance of a vertex, which is determined by the importance of all its neighboring vertices:

$$I(V_i) = (1-d) + d \times \sum_{j \in E(V_i)} \frac{w_{ij}}{\sum_{k \in E(V_j)} w_{jk}} I(V_j) \tag{6}$$

where V_i denotes vertex i, $I(V_i)$ is the importance of V_i, $E(V_i)$ is the set of vertices that are connected to V_i, w_{ij} is the weight of the edge between vertex i and j, d is the damping factor inherited from the PageRank algorithm.

Our graph contains two types of vertices: sentence pair vertices and phrase pair vertices. The importance scores of sentence pair vertices and phrase pair vertices are computed iteratively. Meanwhile, the weight w_{ij} is extended to h_{ij} and g_{ij} as well to reflect the relationships between two types of vertices.

Let $u(s_i)$ and $v(p_j)$ denote the importance scores of a sentence pair vertex s_i and a phrase pair vertex p_j. They are computed by Equations 7 and 8:

$$u(s_i) = (1-d) + d \times \sum_{j \in E_{SP}(s_i)} \frac{h_{ij}}{\sum_{k \in E_{SP}(p_j)} h_{kj}} v(p_j) \tag{7}$$

$$v(p_j) = \alpha \times v_{SP}(p_j) + (1-\alpha) \times v_{PP}(p_j) \tag{8}$$

where $v_{SP}(p_j)$ and $v_{PP}(p_j)$ are the relative confidence contribution from neighboring sentence pair vertices and neighboring phrase pair vertices, which are computed by Equation 9 and 10:

$$v_{SP}(p_j) = (1-d) + d \times \sum_{i \in E_{SP}(p_j)} \frac{h_{ij}}{\sum_{k \in E_{SP}(s_i)} h_{ik}} u(s_i) \tag{9}$$

$$v_{PP}(p_j) = (1-d) + d \times \sum_{i \in E_{PP}(p_j)} \frac{g_{ij}}{\sum_{k \in E_{PP}(p_i)} g_{ik}} v(p_i) \tag{10}$$

where $\alpha \in [0,1]$ is the interpolation factor that can be optimized on the development data; d is set to 0.85, which is the default value in the original PageRank algorithm, $E_{SP}(s_i) = \{j | \langle s_i, p_j \rangle \in E_{SP}\}$, $E_{SP}(p_j) = \{i | \langle s_i, p_j \rangle \in E_{SP}\}$ and $E_{PP}(p_i) = \{j | \langle p_i, p_j \rangle \in E_{PP}\}$.

Based on Equations 7-10, we devise a *Weighted Mutual Recommendation Algorithm* (WMRA) to iteratively compute the importance scores of all the vertices. Let $\mathbf{U} = [u(s_i)]_{n \times 1}$ and $\mathbf{V} = [v(p_j)]_{m \times 1}$ be two vectors denoting the importance scores of all sentence pair vertices and phrase pair vertices, $\mathbf{H} = [h_{ij}]_{n \times m}$ and $\mathbf{G} = [g_{ij}]_{m \times m}$ be the matrixes of the recommendation scores from SPR and PPR, $\hat{\mathbf{H}}$ and $\hat{\mathbf{G}}$ be the corresponding normalized matrixes in which the sum of each row equals to one, and $\tilde{\mathbf{H}}$ be the normalized version of $\mathbf{H}^T$. $\mathbf{I}_n$ is a column vector with n rows and all the elements are one, $\mathbf{V}_{SP} = [v_{SP}(p_j)]_{m \times 1}$ and $\mathbf{V}_{PP} = [v_{PP}(p_j)]_{m \times 1}$.

WMRA is illustrated in Algorithm 1, where $\mathbf{U}$ and $\mathbf{V}$ are initialized to $\mathbf{I}_n$ and $\mathbf{I}_m$ (Lines 1-2). They are iteratively computed and normalized (Lines 7-12), $\mathbf{U}^{(n)}$ and $\mathbf{Q}^{(n)}$ are the results of nth iteration. At the end of each iteration (Lines 13-15), the maximum difference δ of the importance scores between two consecutive iterations is calculated.[1] The above procedure will terminate when δ is lower than a pre-defined threshold (10^{-12} in this study). As shown in Algorithm 1, WMRA is a natural extension of the weighted graph-based random walk in [4]. The difference is that the computation for importance scores of sentence pair vertices and phrase pair vertices is performed individually. In addition, normalization is also conducted separately to guarantee the sum of the importance scores for each type of vertices is equal to one.

2.5 Integration into SMT Log-Linear Model

The importance scores of phrase pairs $\mathbf{V} = [v(t_j)]_{m \times 1}$ produced by WMRA can be directly integrated into the log-linear model as additional new features, which are called the *Phrase Scoring* (PS) features.

[1] For a n dimensional vector $\mathbf{x}$: 1-norm $\|\mathbf{x}\|_1$ equals to $\sum_{i=1}^{n} |x_i|$, while maximum-norm $\|\mathbf{x}\|_\infty$ equals to $\max(|x_1|, |x_2|, \cdots, |x_n|)$.

Algorithm 1. Weighted Mutual Recommendation Algorithm

Require: $\tilde{\mathbf{H}}, \hat{\mathbf{H}}, \hat{\mathbf{G}}$
$\mathbf{U}^{(0)} \leftarrow \mathbf{I}_n$
$\mathbf{V}^{(0)} \leftarrow \mathbf{I}_m$
$\delta \leftarrow$ Infinity
$\epsilon \leftarrow$ threshold
$n \leftarrow 1$
while $\delta > \epsilon$ **do**
  $\mathbf{U}^{(n)} \leftarrow (1-d) \times \mathbf{I}_n + d \times \tilde{\mathbf{H}}^T \times \mathbf{V}^{(n-1)}$
  $\mathbf{V}_{SP}^{(n)} \leftarrow (1-d) \times \mathbf{I}_m + d \times \hat{\mathbf{H}}^T \times \mathbf{U}^{(n-1)}$
  $\mathbf{V}_{PP}^{(n)} \leftarrow (1-d) \times \mathbf{I}_m + d \times \hat{\mathbf{G}}^T \times \mathbf{V}^{(n-1)}$
  $\mathbf{V}^{(n)} \leftarrow \alpha \times \mathbf{V}_{SP}^{(n-1)} + (1-\alpha) \times \mathbf{V}_{PP}^{(n-1)}$
  $\mathbf{U}^{(n)} \leftarrow \frac{\mathbf{U}^{(n)}}{\|\mathbf{U}^{(n)}\|_1}$
  $\mathbf{V}^{(n)} \leftarrow \frac{\mathbf{V}^{(n)}}{\|\mathbf{V}^{(n)}\|_1}$
  $\delta_{\mathbf{U}} \leftarrow \mathbf{U}^{(n)} - \mathbf{U}^{(n-1)}$
  $\delta_{\mathbf{V}} \leftarrow \mathbf{V}^{(n)} - \mathbf{V}^{(n-1)}$
  $\delta \leftarrow \max(\|\delta_{\mathbf{U}}\|_\infty, \|\delta_{\mathbf{V}}\|_\infty)$
  $n \leftarrow n+1$
end while
return $\mathbf{U}^{(n)}, \mathbf{V}^{(n)}$

The importance scores of sentence pair vertices $U = [u(s_i)]_{n\times 1}$ are used as the weights of sentence pairs to re-estimate the probabilities of phrase pairs by MLE method. Followed by the corpus weight estimation approach [6], given a phrase pair $p = \langle \bar{f}, \bar{e} \rangle$, $A(\bar{f})$ and $B(\bar{e})$ indicate the sets of sentences that $\bar{f}$ and $\bar{e}$ occur in. Then the translation probability is defined as:

$$P_{\mathrm{CW}}(\bar{f}|\bar{e}) = \frac{\sum_{i \in A(\bar{f}) \cap B(\bar{e})} u(s_i) \times c_i(\bar{f}, \bar{e})}{\sum_{j \in B(\bar{e})} u(s_j) \times c_j(\bar{e})} \tag{11}$$

where $c_i(\cdot)$ denotes the count of the phrase or phrase pair in s_i. $P_{\mathrm{CW}}(\bar{f}|\bar{e})$ and $P_{\mathrm{CW}}(\bar{e}|\bar{f})$ are called Corpus Weighting (CW) based translation probability, which are also integrated into the log-linear model in addition to the conventional phrase translation probabilities [7].

3 Experiments

3.1 Setup

We evaluated our method on Chinese-to-English machine translation tasks over three experimental settings with different bilingual corpus for domains and sizes.

SLDB+BTEC Setting: A corpus in colloquial style from the DIALOG task of IWSLT 2010 was used, consisting of the Spoken Language Databases (SLDB)

corpus and parts of the Basic Travel Expression Corpus (BTEC) corpus. The Chinese portion contained 655,906 words and the English portion contained 806,833 words. The language model was trained over the English portion of the training corpus. The development dataset was devset8 plus the Chinese DIALOG data set and the test data was devset9.

FBIS Setting: A news domain corpus (FBIS dataset, LDC2003E14) was used in this experiment. The Chinese portion contained 2.99 million words and the English portion contained 3.94 million words. The development dataset was the NIST 2003 evaluation dataset and the test datasets were the NIST 2006 and NIST 2008 evaluation datasets. The language model was trained over the English portion of FBIS plus the Xinhua portion of the Gigaword V4 corpus.

Mixed Domain Setting: This experiment used a mixed-domain bilingual corpus containing around 30 million sentence pairs. The bilingual data was mainly mined from the web, as well as the United Nations parallel corpus released by LDC and the parallel corpus released by China Workshop on Machine Translation (CWMT). The development and test datasets were the same as in the FBIS setting. The language model was trained over the English portion of the bilingual corpus plus the Xinhua portion of the Gigaword V4 corpus.

A phrase-based decoder was implemented based on chart-based CKY parsing with inversion transduction grammar [8]. We used the following feature functions in the log-linear model for the baseline system:

- phrase translation probabilities and lexical weights in both directions (4 features);
- 5-gram language model (1 feature);
- lexicalized reordering model (1 feature);
- phrase count and word count (2 features).

The translation model was trained over the word-aligned bilingual corpus conducted by GIZA++ [9] in both directions, and the diag-grow-final heuristic was used to refine the symmetric word alignment. A 5-gram language model was trained using the modified Kneser-Ney smoothing [10]. The lexicalized reordering model [11] was trained over the parallel data. Case-insensitive BLEU4 [12] was used as the evaluation metric. The parameters of the log-linear model were tuned by optimizing BLEU on the development data using MERT [13]. Statistical significance test was performed using the bootstrap re-sampling method [14].

3.2 Implementation Details

In the baseline system, the phrase pairs that appear only once in the bilingual corpus were simply discarded because most of them were noisy. In addition, the fix-discount method in [1] for phrase table smoothing was also used. This implementation made the baseline system perform much better and the model size was much smaller. In fact, the basic idea of our "one count" cutoff is very

similar to the idea of "leaving-one-out" in [2]. The results in Table 1 show that the "leaving-one-out" method performs almost the same as our baseline, thereby cannot bring other benefits to the system.

When the random walk ran on the mixed-domain bilingual corpora, even filtering phrase pairs that appear only once would still require dozens of days of CPU time for a number of iterations. To overcome this problem, we used a distributed algorithm based on the iterative computation in the Section 2.4. Before the iterative computation starts, the sum of the outlink weights for each vertex was computed first. The edges were randomly partitioned into sets of roughly equal size. Each edge could generate key-value pairs, where the same key were summed locally and accumulated across different machines. Then, in each iteration, the score of each vertex was updated according to the sum of the normalized inlink weights. The algorithm fits well into the MapReduce programming model [15] and we used it as our implementation.

3.3 SMT Performance Evaluation

As mentioned in Section 2.5, we have integrated the new features of PS and CW into the SMT log-linear model as well as the baseline features. This section reports the evaluation results of different settings. The experimental results are shown in Table 1. The results show that WMRA leads to significant performance improvements compared to the baseline, which demonstrates that the recommendation scores propagated among the vertices are quite useful for SMT systems. It seems the integration of PS or CW leads to similar performance improvements, while integrating both of them achieves the best performance.

Table 1. BLEU(%) of Chinese-English translation tasks on three settings ($p < 0.05$)

	SLDB+BTEC	**FBIS**		**Mixed Domain**	
	test	nist2006	nist2008	nist2006	nist2008
Baseline	45.60	31.30	23.29	35.20	29.38
Leaving-one-out	-	-	-	35.30	29.33
+WMRA PS	46.77	32.01	24.13	-	-
+WMRA CW	47.08	32.03	24.10	-	-
+WMRA PS+CW	47.50	32.42	24.77	36.10	30.22

In general, our method improves the BLEU scores significantly over the bilingual corpus of different sizes. The largest improvement comes from the setting of SLDB+BTEC, which improves by 1.9 BLEU points over the baseline. The reason might be that the sentence pairs in SLDB+BTEC are quite similar, so that the graph-based random walk can effectively distinguish the good phrase pairs from the poor ones.

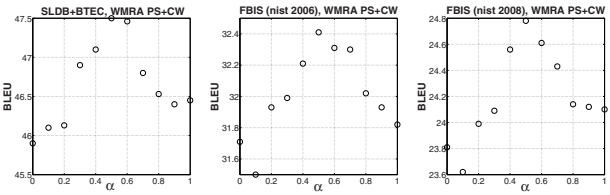

Fig. 3. SMT Performance as the relative contribution ratio α varies. From left to right, the three figures shows the SMT experiments on the *SLDB+BTEC* setting, the FBIS setting with Nist 2006 test set and Nist 2008 test set.

3.4 Interpolation Factor

The interpolation factor α in Equation 8 controls the relative contribution ratio from neighboring sentence pair vertices and neighboring phrase pair vertices. In principle, this ratio should be estimated automatically using machine learning techniques. In order to avoid the overhead of end-to-end training, we empirically tune it in the range of 0 to 1 with an interval of 0.1. Figure 3 shows the tuning results under three different settings when applying WMRA. As shown, the SMT performance arrives at the maximum when α is around 0.5. Therefore, without loss of generality, α is set to 0.5 for all the experiments in Section 3.3.

4 Related Work

4.1 Phrase Scoring and Corpus Weighting

A great deal of work has been done to get high quality translation knowledge and filter out the noise. There are two main categories of approaches addressing these problems. The first category was based on phrase scoring. Some non-parametric Bayesian techniques [16] were used to estimate the weights of phrase pairs. In addition to the generative models, discriminative training for phrase alignment and scoring [17] was also proposed. In this method, the objective function for phrase alignment was optimized jointly with SMT decoding to achieve end-to-end performance improvements. The second category was based on corpus weighting. They tried to handle the problem by corpus weight estimation according to the quality of sentence pairs. A discriminative corpus weighting method [6] was proposed to assign smaller weights to the low quality bilingual sentence pairs. In contrast to previous research, in which corpus weighting and phrase scoring are investigated separately, our method optimizes them collectively and gains more improvements in SMT performance.

4.2 Graph-Based Random Walk

Graph-based random walk was extensively used in web analysis and search. The most famous algorithms were Google's PageRank [3]. Beyond that, graph-based

random walk was also successfully applied to other tasks, such as document summarization, keywords extraction [4] and tags recommendation [18], etc. Moreover, document summarization and keywords extraction were accomplished simultaneously [19, 5], with the graph being built using the relationships between sentences and words homogenously and heterogeneously. Recently, graph-based random walk has been used for SMT to clean the noisy bilingual data [20], which can be considered as an unsupervised approach for corpus weighting. Inspired by previous work, our method uses the graph-based random walk to distinguish high quality translation knowledge from noise, which is better than the traditional MLE approach for the parameter estimation.

5 Conclusion and Future Work

In this paper, we have developed an effective approach to optimize phrase scoring and corpus weighting jointly using graph-based random walk. The proposed approach automatically estimates the quality of parallel sentence pairs and phrase pairs by performing mutual recommendation. We convert the importance scores into new features and integrate them into the log-linear model of the SMT system. Significant improvements are achieved in our experiments.

In the future, we will extend our method to other SMT models such as the hierarchical phrase-based model and syntax-based models in which non-terminals are contained in syntactic translation rules. These extensions have higher complexity because more translation rules will be extracted from the bilingual corpus. To this end, we will further optimize our algorithm based on the divide-and-conquer strategy when the graph size is extremely large.

References

[1] Foster, G., Kuhn, R., Johnson, H.: Phrasetable smoothing for statistical machine translation. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp. 53–61. Association for Computational Linguistics, Sydney (2006)

[2] Wuebker, J., Mauser, A., Ney, H.: Training phrase translation models with leaving-one-out. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 475–484. Association for Computational Linguistics, Uppsala (2010)

[3] Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. Computer Networks and ISDN Systems 30(1), 107–117 (1998)

[4] Mihalcea, R., Tarau, P.: Textrank: Bringing order into texts. In: Lin, D., Wu, D. (eds.) Proceedings of EMNLP 2004, pp. 404–411. Association for Computational Linguistics, Barcelona (2004)

[5] Wan, X., Yang, J., Xiao, J.: Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 552–559. Association for Computational Linguistics, Prague (2007)

[6] Matsoukas, S., Rosti, A.V.I., Zhang, B.: Discriminative corpus weight estimation for machine translation. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp. 708–717. Association for Computational Linguistics, Singapore (2009)
[7] Koehn, P., Och, F.J., Marcu, D.: Statistical phrase-based translation. In: Proceedings of HLT-NAACL 2003 Main Papers, pp. 48–54. Association for Computational Linguistics, Edmonton (2003)
[8] Wu, D.: Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. Computational Linguistics 23(3), 377–403 (1997)
[9] Och, F.J., Ney, H.: A systematic comparison of various statistical alignment models. Computational Linguistics 29(1), 19–51 (2003)
[10] Kneser, R., Ney, H.: Improved backing-off for m-gram language modeling. In: 1995 International Conference on Acoustics, Speech, and Signal Processing, ICASSP 1995, vol. 1, pp. 181–184. IEEE (1995)
[11] Xiong, D., Liu, Q., Lin, S.: Maximum entropy based phrase reordering model for statistical machine translation. In: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 521–528. Association for Computational Linguistics, Sydney (2006)
[12] Papineni, K., Roukos, S., Ward, T., Zhu, W.-J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of 40th Annual Meeting of the Association for Computational Linguistics, pp. 311–318. Association for Computational Linguistics, Philadelphia (2002)
[13] Och, F.J.: Minimum error rate training in statistical machine translation. In: Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp. 160–167. Association for Computational Linguistics, Sapporo (2003)
[14] Koehn, P.: Statistical significance tests for machine translation evaluation. In: Lin, D., Wu, D. (eds.) Proceedings of EMNLP 2004, pp. 388–395. Association for Computational Linguistics, Barcelona (2004)
[15] Dean, J., Ghemawat, S.: Mapreduce: simplified data processing on large clusters. Communications of the ACM 51(1), 107–113 (2008)
[16] DeNero, J., Bouchard-Côté, A., Klein, D.: Sampling alignment structure under a Bayesian translation model. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pp. 314–323. Association for Computational Linguistics, Honolulu (2008)
[17] Deng, Y., Xu, J., Gao, Y.: Phrase table training for precision and recall: What makes a good phrase and a good phrase pair? In: Proceedings of ACL 2008: HLT, pp. 81–88. Association for Computational Linguistics, Columbus (2008)
[18] Guan, Z., Bu, J., Mei, Q., Chen, C., Wang, C.: Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, pp. 540–547. ACM, New York (2009)
[19] Zha, H.: Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2002, pp. 113–120. ACM, New York (2002)
[20] Cui, L., Zhang, D., Liu, S., Li, M., Zhou, M.: Bilingual data cleaning for smt using graph-based random walk. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics. Short Papers, vol. 2, pp. 340–345. Association for Computational Linguistics, Sofia (2013)