Dynamically Integrating Cross-Domain Translation Memory into Phrase-Based Machine Translation during Decoding

Kun Wang[†] **Chengqing Zong**[†] **Keh-Yih Su**[‡]
[†]National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, China
[‡]Institute of Information Science, Academia Sinica, Taiwan
[†]{kunwang, cqzong}@nlpr.ia.ac.cn
[‡]kysu@iis.sinica.edu.tw

Abstract

Our previous work focuses on combining translation memory (TM) and statistical machine translation (SMT) when the TM database and the SMT training set are the same. However, the TM database will deviate from the SMT training set in the real task when time goes by. In this work, we concentrate on the task when the TM database and the SMT training set are different and even from different domains. Firstly, we dynamically merge the matched TM phrase-pairs into the SMT phrase table to meet the real application. Secondly, we propose an improved integrated model to distinguish the original and the newly-added phrase-pairs. Thirdly, a simple but effective TM adaptation method is adopted to favor the consistent translations in cross-domain test. Our experiments have shown that merging the TM phrase-pairs achieves significant improvements. Furthermore, the proposed approaches are significantly better than the TM, the SMT and previous integration works for both in-domain and cross-domain tests.

1 Introduction

Since the translation memory (TM) system and the statistical machine translation (SMT) system complement each other in those matched sub-segments and unmatched sub-segments (Wang et al., 2013), combining them can improve the output quality significantly, especially when high-similarity fuzzy matches are available. Therefore, combining TM and SMT is drawing more and more attention in recent years (He et al., 2010a; 2010b; 2011; Koehn and Senellart, 2010; Zhechev and van Genabith, 2010; Ma et al., 2011; Dara et al., 2013; Wang et al., 2013).

Those previous works on combining TM and SMT can be classified into four categories: (1) selecting the better translation sentence from TM and SMT (He et al., 2010a; 2010b; Dara et al., 2013); (2) incorporating TM matched sub-segments into SMT in a pipelined manner (Koehn and Senellart, 2010; He et al., 2011; Ma et al., 2011); (3) only enhancing the SMT phrase table with new TM phrase-pairs (Bi çici and Dymetman, 2008; Simard and Isabelle, 2009); and (4) incorporating the associated TM information with each source phrase to guide the SMT decoding (Wang et al., 2013).

However, all previous works mentioned above only focus on the case in which the TM database and the SMT training set share the same data-set. Nonetheless, in real applications, the TM database will deviate from the SMT training set when time goes by, because the TM database will be dynamically enlarged when more translations are generated by the human translator. Therefore, this paper will concentrate on a more realistic case, in which the TM database and the SMT training set are different and even from different domains.

When the TM database and the SMT training set share the same data-set, the integrated model (Wang et al., 2013) can avoid the drawbacks of the pipeline approaches and outperforms the other approaches significantly. However, this integrated model only refers to the TM information but not adopts the matched TM phrase-pairs as candidates during decoding. Therefore, many TM phrase-pairs cannot be covered by the SMT phrase table when the TM database and the SMT training set are dif-

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 398–408, Dublin, Ireland, August 23-29 2014.

ferent. It is thus impossible to generate those unseen TM target phrases. This problem would even get worse when the TM database and the SMT training set are from different domains.

To make the integrated model meet the real application, we dynamically merge the matched TM phrase-pairs into the SMT phrase table. In addition, an improved integrated model is proposed to distinguish the original SMT phrase-pairs and the newly-added ones extracted from TM. Furthermore, a simple but effective TM adaptation method is adopted to favor the consistent translation in cross-domain test. To our best knowledge, this is the first unified framework for integrating TM into SMT during decoding when the TM database and the SMT training set are different (even from different domains).

On the TM database which consists of Chinese–English computer technical documents, our experiments have shown that merging the matched TM phrase-pairs achieves significant improvement when the fuzzy match score is above 0.5. Besides, the proposed approaches are significantly better than either the SMT or the TM systems for both the in-domain and the cross-domain tests when the fuzzy match score is above 0.4. Furthermore, the proposed approaches also outperform previous integration works significantly in all test conditions.

2 Integrated Model

Wang et al. (2013) incorporated the TM information into the phrase-based SMT, and re-defined the translation problem as:

$$\hat{t} = \arg\max_t P(t|s, tm\_s, tm\_t, tm\_f, s\_a, tm\_a)$$

Where s denotes the given source sentence, t is a corresponding target translation, and $\hat{t}$ is the final result; $[tm\_s, tm\_t, tm\_f, s\_a, tm\_a]$ is the associated information of the best TM sentence-pairs; tm_s and tm_t are the corresponding TM source and target sentences, respectively; tm_f denotes its corresponding fuzzy match score (from 0 to 1); s_a is the monolingual alignment information between s and tm_s; and tm_a denotes the bilingual word alignment information between tm_s and tm_t.

With the TM information, this problem can be simplified to:

$$\hat{t} \triangleq \text{argmax}\left\{P\left(\bar{t}_1^{\text{K}}\middle|\bar{s}_{a(1)}^{a(\text{K})}\right) \times \prod_{k=1}^{K} \max_{tm\_\bar{t}_{a(k)}} P(M_k|L_k, z)\right\} \quad (1)$$

Where $\bar{s}_{a(k)}$ and $\bar{t}_k$ denote the k-th associated source and target phrases, respectively; $tm\_\bar{s}_{a(k)}$ and $tm\_\bar{t}_{a(k)}$ are the corresponding TM source and target phrases associated with the given source phrase $\bar{s}_{a(k)}$ (total K phrases without insertion). M_k is the corresponding TM target phrase matching status for the current target candidate $\bar{t}_k$, which reflects the quality of the given candidate; L_k is the linking status vector of $\bar{s}_{a(k)}$ (the aligned source phrase, within $\bar{s}_{a(1)}^{a(\text{K})}$, of $\bar{t}_k$), which indicates the matching and linking status in the source side (and is closely related to the matching status of the target side). tm_f is uniformly divided into ten fuzzy match intervals and the index z specifies the corresponding interval.

In Equation (1), the first factor is just the typical phrase-based SMT model, and the second factor $P(M_k|L_k, z)$ is the information derived from the TM sentence pair. Afterwards, the factor $P(M_k|L_k, z)$ was further derived with TM matching status as follows:

$$P(M_k|L_k, z) \approx \begin{Bmatrix} \text{P}(TCM_k|SCM_k, NLN_k, LTC_k, SPL_k, SEP_k, z) \\ \times \text{P}(LTC_k|CSS_k, SCM_k, NLN_k, SEP_k, z) \\ \times \text{P}(CPM_k|TCM_k, SCM_k, NLN_k, z) \end{Bmatrix} \quad (2)$$

Where the first factor reflects the TM content matching status, the second factor is the relationship between various TM target phrases, and the third factor is the reordering information implied by TM. Equation (2) is adopted to guide the SMT decoding, and is denoted as the integrated Model-III in (Wang et al., 2013) (also called **Model-III** in this paper thereafter).

For space limitation, only those features which are also adopted in our additional introduced probability factor (to be specified later) will be briefly introduced here:

Target Phrase Content Matching Status (TCM): It indicates the content matching status between $\bar{t}_k$ and $tm\_\bar{t}_{a(k)}$, and reflects the quality of $\bar{t}_k$. It is a member of {*Same*, *High*, *Low*, *NA* (*Not-Applicable*)}.

Source Phrase Content Matching Status (SCM): It indicates the content matching status between $\bar{s}_{a(k)}$ and $tm\_\bar{s}_{a(k)}$, and affects the matching status of $\bar{t}_k$ and $tm\_\bar{t}_{a(k)}$ greatly. It is a member of {*Same*, *High*, *Low*, *NA*}.

Number of Linking Neighbors (NLN): Usually, the context of a source phrase would affect its target translation. The more similar the context is, the more likely that the translation is the same. NLN is adopted to measure the context similarity.

3 Proposed Approaches

3.1 Merging the TM Phrase-Pairs

Since all TM phrase-pairs are only referred while re-scoring the SMT candidates in Model-III, they are not regarded as candidates during decoding. When the TM database and the SMT training set are the same, this restriction is reasonable because the SMT phrase table can cover all the continuous TM phrase pairs within the phrase length limit. However, this would not be true when the TM database and the SMT training set are different. Therefore, the SMT phrase table should be further enhanced with those matched new TM phrase pairs in this case.

According to their relations with the SMT phrase table, TM phrase pairs can be classified into three different categories: (1) the whole TM phrase-pair can be found in the original SMT phrase table; (2) only TM source phrase exists in the original SMT phrase table, but its corresponding target phrase does not; (3) even TM source phrase cannot be found in the original SMT phrase table. Since the first category has been covered by the original SMT phrase table, only the phrase-pairs from the second and the third categories should be added into the SMT phrase table dynamically for each input sentence. To distinguish those newly added phrase-pairs from the original SMT phrase-pairs, we use eight additional feature weights λ_m for the translation probability (lexical and phrase transfer in both directions) and two more feature weights for the phrase penalty (details will be specified later in Section 4).

The above approach is inspired by the work of (Bi ɕci and Dymetman, 2008). However, there are three differences between our approach and theirs. Firstly, we add all those matched TM phrase-pairs (include all associated sub-phrase pairs), while Bi ɕci and Dymetman (2008) only added the longest matched one; Secondly, we add all the possible TM target phrase-pairs for a given TM source phrase while they extracted only one TM target phrase regardless of the existence of multiple TM target candidates; Lastly, we use different feature weights to distinguish those newly added TM phrase-pairs from the original SMT phrase-pairs, while they treated them equally.

3.2 Distinguishing the TM Phrase-Pairs

As mentioned in Section 3.1, we need to merge those TM matched phrase pairs into the SMT phrase table when the TM database and the SMT training set are different. However, the original integrated Model-III does not distinguish the newly added TM phrase-pairs from those original SMT phrase-pairs in $P(M_k|L_k,z)$. Therefore, we introduce two new features ***Source Phrase Origin*** (**SPO**) and ***Target Phrase Origin*** (**TPO**), which are a member of {*Original*, *Newly-Added*}, to the original Model-III in (Wang et al., 2013) to favor the newly added TM phrase-pairs, and re-derive $P(M_k|L_k,z)$ as follows (assume that TPO is only dependent on SPO, NLN and z):

$$\begin{aligned} & P(M_k|L_k,z) \\ & \triangleq P([TCM,LTC,CPM,TPO]_k|[SCM,NLN,CSS,SPL,SEP,SPO]_k,z) \\ & \approx \left\{ \begin{array}{l} \mathrm{P}(TCM_k|SCM_k,NLN_k,LTC_k,SPL_k,SEP_k,z) \\ \quad \times \mathrm{P}(LTC_k|CSS_k,SCM_k,NLN_k,SEP_k,z) \\ \quad \times \mathrm{P}(CPM_k|TCM_k,SCM_k,NLN_k,z) \\ \quad \times \mathrm{P}(TPO_k|SPO_k,NLN_k,z) \end{array} \right\} \end{aligned} \quad (2)$$

The additional factor $\mathrm{P}(TPO_k|SPO_k,NLN_k,z)$ in the above equation is added to handle those newly added TM phrase-pairs. This would be the proposed ***Distinguishing Model***. For the phrases from the original SMT phrase table, both the SPO and TPO features would be "*Original*"; for the phrases from the second category mentioned in Section 3.1, the SPO would be "*Original*" but the TPO would be "*Newly-Added*"; for the phrases from the third category, both the SPO and TPO features would be "*Newly-Added*".

3.3 TM Adaptation

In real applications, the TM database is usually not big enough to train an SMT system when it is applied to a special technical domain other than the news domain. Besides, many professional translators do not want to expose the whole TM database to the SMT system providers (Cancedda, 2012). In this situation, we will be forced to first train an SMT model on an **out** domain (usually the news domain) which possesses a lot of training data, and then fix the obtained phrase-based SMT model. Afterwards, we incorporate it on line with an additional TM database which is from another **in** domain.

To simulate the above scenario, we will thus train our integrated model on the out domain. However, we have a domain-mismatch problem for this cross-domain test. Generally, in the technical domain, which is suitable for TM application, the translations (especially for technical terms) are much more consistent than that in the news domain. That is, the same source phrase in various places tends to have exactly the same translation in technical domains. Therefore, when we use Distinguishing Model to perform forced decoding, the obtained results would possess different statistics among the in-domain development set and the out-domain training set. For example, at interval [0.9, 1.0), when SCM is "Same", 94.6% of TCM are "Same" in the development set (**in**), while this ratio is only 65.1% in the training set (**out**). Therefore, the factor $P(TCM_k|SCM_k, NLN_k, LTC_k, SPL_k, SEP_k, z)$ from the test set will possess a different probability distribution in comparison with that from the training set. However, the development set is not big enough (only a few hundreds sentence-pairs at each interval) to re-train all TM factors of the proposed model. Therefore, we simply add the following h_1 feature to reflect the tendency of having high translation consistency in the development set:

$$h_1(\bar{t}, \bar{s}, z) = \begin{cases} 1.0, if\ SCM_k = Same \text{ and } TCM_k = Same \\ 0.0, otherwise \end{cases}$$

Where $\bar{s}$ and $\bar{t}$ denote the source phrase, the target candidate, respectively.

Furthermore, various source synonyms might generate the same translation (Zhu et al., 2013). Therefore, even SCM≠Same, we still favor the SMT phrase-pair candidate which exactly matches TM target phrase. For example, if source words are synonyms such as "需要" (want) and "要" (want), "如果" (if) and "若" (if), "立即" (at once) and "马上" (at once), the target translations would be the same. Therefore, the issue of having high translation consistency in the technical domain is also applied. We thus further add the following h_2 feature to reflect the tendency of having high translation consistency in this case ("High" and "Low" are grouped into "Other" for the SCM):

$$h_2(\bar{t}, \bar{s}, z) = \begin{cases} 1.0, if\ SCM_k = Other \text{ and } TCM_k = Same \\ 0.0, otherwise \end{cases}$$

Afterwards, the associated feature weights are tuned on the development set.

4 Experiments

4.1 Experimental Setup

We use the same TM data-set adopted by Wang et al. (2013), which is a Chinese–English TM database consisting of computer technical documents. It includes about 267k sentence pairs. All the experiments are conducted around this TM data-set. To compare the performances under different conditions, the same development set and the test set will be shared by both in-domain and cross-domain tests. Since the associated SMT training-set and TM database will vary under different experimental configurations, they will be specified later in each sub-section.

In this work, the translation memory system (denoted as **TM**) and the phrase-based machine translation system (denoted as **SMT**) are adopted as our two baseline systems. Following (Wang et al., 2013), for TM, the word-based fuzzy match score is adopted as the similarity measure; also, for the phrase-based SMT system, the same Moses toolkit (Koehn et al., 2007) and the same set of following features are adopted: the phrase translation model, the language model, the distance-based reordering model, the lexicalized reordering model and the word penalty. The system configurations are as follows: GIZA++ (Och and Ney, 2003) is used to obtain the bidirectional word alignments. Afterwards, "intersection" refinement (Koehn et al., 2003) is adopted to extract phrase-pairs. We use SRI Language Model

	#Sentences	#Chn. Words	#Chn. VOC.	#Eng. Words	#Eng. VOC.
New TM Database	130,953	1,808,992	30,164	1,811,413	30,807
SMT Training Set	130,953	1,814,524	29,792	1,815,615	30,516

Table 1: Corpus Statistics for In-Domain Tests

Intervals	[0.9, 1.0)	[0.8, 0.9)	[0.7, 0.8)	[0.6, 0.7)	[0.5, 0.6)	[0.4, 0.5)	[0.3, 0.4)	(0.0, 0.3)	**(0.0, 1.0)**
#Sentences	147	255	244	355	488	514	419	154	**2,576**
#Words	2,431	3,438	3,299	4,674	6,125	7,525	7,082	4,074	**38,648**
W/S	16.5	13.5	13.5	13.2	12.6	14.6	16.9	26.5	**15.0**

Table 2: Corpus Statistics for In-Domain Test-Set (W/S: the average #words per sentence)

toolkit (Stolcke, 2002) to train a 5-gram model with modified Kneser-Ney smoothing (Kneser and Ney, 1995; Chen and Goodman, 1998) on the target-side (English) training corpus. All the feature weights and the weight for each probability factor are tuned on the development set with minimum-error-rate training (MERT) (Och, 2003). The maximum phrase length is set to 7 in our experiments.

To compare our proposed models with those state-of-the-art methods, we re-implement two XML-Markup approaches (Koehn and Senellart, 2010; and the upper bound version of (Ma et al, 2011)) and the Model-III (Wang et al., 2013) as three baseline systems, and denote them as **Koehn-10**, **Ma-11-U** and **Model-III**, respectively. Similar to (Wang et al., 2013), we only re-implement the XML-Markup method used in (Ma et al, 2011), but not their discriminative learning method.

Following (Wang et al., 2013), we also train the TCM, LTC and CPM factors in the SMT training set with cross-fold translation. Since the TPO factor (conditioning on NLN and Distinguishing Model) is based on Model-III, we first use Model-III to generate the desired results on the development set via forced decoding, and then generate the training samples of TPO factor for Distinguishing Model.

In this work, the translation performance is measured with case-insensitive BLEU-4 score (Papineni et al., 2002) and TER score (Snover et al., 2006). Statistical significance tests are conducted with re-sampling (1,000 times) approach (Koehn, 2004) in 95% confidence level.

4.2 In-Domain Translation Results

In the in-domain test, the original TM dataset is first randomly divided into two parts. The first part is then adopted as the new TM database, while the second part is adopted as the SMT training set. The detailed corpus statistics is shown in Table 1. Since the TM database is different from that adopted in (Wang et al., 2013), the statistics shown in Table 2 at each interval is also different from theirs.

All matched TM phrase-pairs are extracted according to the word alignment generated from the phrase-based SMT system. Since there are not enough samples to estimate the translation probabilities for those newly added TM phrase-pairs, we use the following method to assign the translation probabilities. For those TM phrase-pairs that only their source phrases exist in the original SMT phrase table (the second category mentioned in Section 3.1), as their source phrases have already existed in the SMT phrase table, there is at least one associated target phrase in the original SMT phrase table. For each new TM phrase-pair, we thus directly assign the maximum probability among its associated original target phrases to it. For those TM phrase-pairs that even their source phrase cannot be found in the original SMT phrase table (the third category), as there is no corresponding phrase-pair in the original SMT phrase table, we will simply assign probability "1.0" (this value is not important as its associated weight will be tuned later) as their four translation probabilities. To distinguish those newly added phrase-pairs from the original SMT phrase-pairs, we use eight additional feature weights for the translation probability and two more feature weights for the phrase penalty.

To evaluate the effectiveness of adding TM phrase-pairs, we compare the cases of whether merging TM phrase-pairs or not for both SMT and Model-III. Table 3 and Table 4 give the translation results in BLEU and TER, respectively. "SMT" and "Model-III" denote that we do not merge the TM phrase-pairs into the SMT phrase table during decoding. That is, they only use the original SMT phrase table.

Intervals	TM	SMT	SMT$^+$	Model-III	Model-III$^+$	Distinguishing	Koehn-10	Ma-11-U
[0.9, 1.0)	79.89	63.65	73.55 +	80.69	86.40 +*#	**86.69** +*#	82.21	67.58
[0.8, 0.9)	72.65	60.75	74.04 +	78.95 *	83.35 +*#	**83.44** +*#	79.50 *	67.03
[0.7, 0.8)	59.59	60.57	65.52 +	68.55 *	71.37 +*#	**72.06** +*#	67.52	62.60
[0.6, 0.7)	41.57	53.38	56.14 +	55.61 #	57.75 +*#	**58.73** +*#$	51.83	56.74
[0.5, 0.6)	25.17	45.60	46.95 +	47.40 #	**48.39** +*#	48.27 *#	39.08	47.94
[0.4, 0.5)	14.62	41.81	42.03	42.60 #	42.30 #	**43.04** *#$	31.60	42.93
[0.3, 0.4)	7.50	**35.95**	35.49	36.10 #	35.31 #	35.34 #	25.25	36.58
(0.0, 0.3)	4.94	32.64	33.22	33.45 #	**33.23** #	**33.23** #	23.70	33.10
(0.0, 1.0)	31.11	46.68	49.41 +	51.00 *#	52.26 +*#	**52.56** +*#$	44.28	48.91

Table 3: In-Domain Translation Results (BLEU). Scores marked with "+" indicates that those newly added TM phrase-pairs significantly ($p < 0.05$) improve the translation results ("SMT" vs. "SMT$^+$", "Model-III" vs. "Model-III$^+$", and "Model-III" vs. "Distinguishing"). Scores marked with "*" are significantly better ($p < 0.05$) than both TM and SMT$^+$ systems, and those marked with "#" are significantly better ($p < 0.05$) than Koehn-10. Scores marked with "$" are significantly better ($p < 0.05$) than Model-III$^+$ ("Model-III$^+$" vs. "Distinguishing")

Intervals	TM	SMT	SMT$^+$	Model-III	Model-III$^+$	Distinguishing	Koehn-10	Ma-11-U
[0.9, 1.0)	10.42	27.14	17.64 +	13.32	8.76 +*#	**8.22** +*#	12.95	23.94
[0.8, 0.9)	16.07	28.73	17.66 +	14.69 *	**10.46** +*#	10.49 +*#	14.72 *	23.83
[0.7, 0.8)	28.68	29.47	24.99 +	22.01 *	20.15 +*#	**19.33** +*#	23.96	27.43
[0.6, 0.7)	48.59	33.76	31.53 +	31.57 #	29.77 +*#	**28.95** +*#$	36.89	30.98
[0.5, 0.6)	63.13	40.57	39.00 +	38.79 #	**38.00** *#	38.51 #	47.08	38.44
[0.4, 0.5)	74.02	44.09	43.66	**42.84 *#**	43.43 #	42.88 *#$	55.35	42.31
[0.3, 0.4)	81.09	**50.00**	50.63	50.04 #	50.70 #	50.90 #	63.28	48.83
(0.0, 0.3)	84.34	**55.58**	56.66	54.68 #	55.96 *#	55.96 *#	68.00	54.51
(0.0, 1.0)	58.58	40.88	38.55 +	37.26 *#	36.47 +*#	**36.28** +*#	45.63	38.73

Table 4: In-Domain Translation Results (TER). The marks are the same as that in Table 3.

"SMT$^+$" and "Model-III$^+$" mean that we merge the TM phrase-pairs into the SMT phrase table dynamically. In these tables, "+" indicates that those newly added TM phrase-pairs significantly improve the translation results ("SMT" vs. "SMT$^+$", "Model-III" vs. "Model-III$^+$", and "Model-III" vs. "Distinguishing").

It can be seen that adding TM phrase-pairs significantly improve the translation results when the fuzzy match score is above 0.5 (comparing SMT with SMT$^+$, and Model-III with Model-III$^+$). For example, at interval [0.9, 1.0), those added TM phrase-pairs significantly improve the SMT system from 63.65 to 73.55, and Model-III from 80.69 to 86.40. However, if Model-III$^+$ is compared with Model-III, the improvements from merging the TM phrase-pairs get less when the fuzzy match score decreases, because the matched TM parts are fewer at low fuzzy match intervals.

Also, with the same original SMT phrase table, Model-III exceeds the SMT system at each interval. For example, at interval [0.9, 1.0), the TM information significantly improve the translation result from 63.65 to 80.69. It thus shows that the TM information is very useful. However, it is still worse than the TM in TER (13.32 vs. 10.42). On the other hand, although Model-III has greatly exceeded the SMT at each interval, Model-III$^+$ still significantly outperforms Model-III at most intervals. Therefore, the benefit of utilizing TM information and the benefit of adding TM phrase-pairs are not covered by each other and can be jointly enjoyed. Take the interval [0.9, 1.0) as an example, the TM information first improve the translation results from 63.65 (SMT) to 80.69 (Model-III), and then the added TM phrase-pairs further boosts it to 86.40 (Model-III$^+$).

Besides, Table 3 and Table 4 also present the translation results of our other two baselines (Koehn-10 and Ma-11-U), and the proposed Distinguishing Model. Scores marked with "*" indicate that they are significantly better ($p < 0.05$) than both the TM and the SMT+ baselines, and those marked with "#" are significantly better ($p < 0.05$) than Koehn-10. Scores marked with "$" are significantly better than Model-III$^+$. The bold entries are the best result at each interval.

In comparison with the TM and the SMT$^+$ systems, Model-III$^+$ is significantly better than both of them in either BLEU or TER scores when the fuzzy match score is above 0.5; also, Distinguishing Model outperforms both the TM and the SMT$^+$ systems in either BLEU or TER scores when the fuzzy match score is above 0.4. Furthermore, the improvements from both Model-III$^+$ and Distinguishing Model get less when the fuzzy match score decreases, as the TM information is less reliable at low fuzzy match intervals.

Across all intervals (the last row in the table), Distinguishing Model not only achieves the best BLEU score (52.56), but also gets the best TER score (36.28). At those intervals when the fuzzy match score is above 0.4, Model-III$^+$ and Distinguishing Model are the best two in either BLEU or TER scores. Besides, Distinguishing Model slightly exceeds Model-III$^+$ at most intervals. However, both Model-III$^+$ and Distinguishing Model achieve significant improvements over the TM and the SMT$^+$.

Compared with previous works, it can be seen that both Model-III$^+$ and Distinguishing Model significantly outperform Koehn-10 in either BLEU or TER scores at all intervals, and are significantly better than Model-III when the fuzzy match score is above 0.6. Furthermore, the proposed approaches (both Model-III$^+$ and Distinguishing Model) achieve a much better TER score than the TM system does at the interval [0.9, 1.0); while Model-III and Koehn-10 are worse than the TM system at this interval. Also, both Model-III$^+$ and Distinguishing Model exceed Ma-11-U at most intervals. Therefore, it can be concluded that the proposed models outperform previous approaches significantly in this scenario.

To further verify the proposed approaches in this case, we swap the TM database and the SMT training set and re-run the experiments. Similar and significant improvements are still observed: both Model-III$^+$ and the Distinguishing Model achieve significant improvements over the TM and the SMT$^+$. All those results have shown that the proposed approaches are robust.

In real environments, the SMT training set and the TM database could be the same before translation projects starts. However, the TM database will gradually deviate from the SMT training set while the translation task progresses. Nonetheless, our experiments have shown that the proposed Distinguishing Model is effective even when the TM database and the SMT training set are totally different (which would be the extreme case for real applications). Therefore, it can be concluded that this proposed approach is robust.

4.3 Cross-Domain Translation Results

To evaluate the cross domain performance, we adopt the news corpora about computer and science from CWMT09 (Liu and Zhao, 2009) as the SMT training set, and adopt the whole TM dataset as the TM database. The SMT training set includes about 404k bilingual sentence-pairs (which includes about 9M Chinese words and 8.7M English words). Corpus statistics is shown in Table 5. Since the TM database and the test set (also the development set) are the same as that in (Wang et al., 2013), the statistics at each interval is the same as theirs but different from Table 2.

The training procedure is the same as that mentioned in the last sub-section. Table 6 and Table 7 present the translation results of TM, SMT, SMT$^+$, two baselines (Koehn-10 and Model-III), and three proposed approaches (Model-III$^+$, Distinguishing and Adaptation). The Adaptation approach means that we add two consistent related features based on Distinguishing Model (Section 3.3). All the formats are the same as that adopted in Table 3 and Table 4. Besides, scores marked by "&" are significantly better than Distinguishing Model.

Comparing the TM with the SMT, the performance of in-domain TM significantly exceeds that of out-domain SMT. Since the fuzzy match intervals are divided according to the TM database, the translation result of the SMT system at interval [0.8, 0.9) even slightly outperforms that at interval [0.9, 1.0). Besides, adding TM phrase-pairs significantly improves the translation results when the fuzzy match score is above 0.5 (SMT vs. SMT$^+$, and Model-III vs. Model-III$^+$). Furthermore, the benefit of utilizing TM information and the benefit of adding TM phrase-pairs are not covered by each other, and can be jointly enjoyed. Furthermore, compared with TM, SMT, SMT$^+$ and Model-III, both Model-III$^+$ and Distinguishing Model achieve better translation results when the fuzzy match score is above 0.4. All observed trends are similar to that in the last sub-section.

	#Sentences	#Chn. Words	#Chn. VOC.	#Eng. Words	#Eng. VOC.
TM Database	261,906	3,623,516	43,112	3,627,028	44,221
SMT Training Set	404,172	9,007,614	102,073	8,737,801	107,883

Table 5: Corpus Statistics for Cross-Domain Tests

Intervals	TM	SMT	SMT$^+$	Model-III	Model-III$^+$	Distinguishing	Adaptation	Koehn-10
[0.9, 1.0)	81.31	30.87	64.74 +	64.79	82.28 +	83.19 +*$	**84.89** *#$&	81.52
[0.8, 0.9)	73.25	31.94	60.13 +	61.91	74.21 +	74.72 +*	**79.78** *#$&	76.47 *
[0.7, 0.8)	63.62	30.63	51.64 +	51.44	62.94 +	63.32 +	**67.74** *$&	67.12 *$&
[0.6, 0.7)	43.64	28.95	39.94 +	38.28	46.28 +*	46.46 +*	**49.49** *$&	48.47 *
[0.5, 0.6)	27.37	27.61	32.49 +	28.85	34.50 +*	34.87 +*	**37.12** *#$&	35.25 *
[0.4, 0.5)	15.43	27.16	27.35	27.30 #	27.47 #	27.82 #	**28.80** *#$&	25.10
[0.3, 0.4)	8.24	**23.85**	22.66	23.81 #	22.41 #	22.41 #	22.95 #	20.72
(0.0, 0.3)	4.13	**24.64**	24.25	24.24 #	23.65 #	24.12 #	24.31 #	18.79
(0.0, 1.0)	40.17	28.30	40.59 +	40.47	47.37 +*	47.70 +*#$	**49.79** *#$&	47.09 *

Table 6: Cross-Domain Translation Results (BLEU). The marks are the same as that in Table 3. Besides, scores marked by "$" are significantly better ($p < 0.05$) than Model-III$^+$, and those marked by "&" are significantly better than "Distinguishing" ("Adaptation" vs. "Distinguishing").

Intervals	TM	SMT	SMT$^+$	Model-III	Model-III$^+$	Distinguishing	Adaptation	Koehn-10
[0.9, 1.0)	9.79	54.54	27.07 +	27.09	11.81 +	11.01 +	**9.58** #$&	13.51
[0.8, 0.9)	16.21	52.86	29.33 +	28.04	17.13 +	17.47 +	**13.80** *#$&	17.29
[0.7, 0.8)	27.79	52.42	36.48 +	35.56	27.07 +	26.40 +$	**23.04** *$&	24.31 *$&
[0.6, 0.7)	46.40	54.74	47.39 +	48.06	41.13 +*	40.36 +*$	**37.45** *#$&	40.16 *
[0.5, 0.6)	62.59	57.18	53.08 +	56.78	51.77 +*	51.60 +*	**48.08** *#$&	51.57
[0.4, 0.5)	73.93	57.19	56.57	57.19 #	56.82 #	56.53 #	**54.42** *#$&	61.32
[0.3, 0.4)	79.86	60.62	61.16	61.35 #	61.31 #	61.31 #	**60.33** #$&	68.82
(0.0, 0.3)	85.31	63.62	62.81	62.22 #	63.04 #	62.07 #	**61.87** #	74.85
(0.0, 1.0)	50.51	56.42	46.89 +	47.38 #	41.63 +*#	41.27 +*#$	**38.87** *#$&	43.95 *

Table 7: Cross-Domain Translation Results (TER). The marks are the same as that in Table 6.

However, both Model-III$^+$ and Distinguishing Model are worse than Koehn-10 at some high fuzzy match intervals. The reason is that the TM factors are trained on the news domain but the test set is from computer technical domain. Therefore, it is not strange that the Adaptation approach achieves the best translation results at all intervals in either BLEU or TER when the fuzzy match score is above 0.4. At most intervals, the Adaptation approach significantly outperforms Koehn-10 in either BLEU or TER, especially for the high fuzzy match intervals such as [0.9, 1.0) and [0.8, 0.9). Furthermore, the Adaptation approach achieves better TER than the TM system and Koehn-10 at intervals [0.9, 1.0) and [0.8, 0.9). All obtained results have shown that the Adaptation approach is effective and robust for cross-domain test. Moreover, it can be seen that the h1 feature (mentioned in Section 3.3) is more effective than the h2 feature.

5 Related Work

According to the way of combination, those previous works can be classified into four categories (as specified in Section 1). The first category uses a classifier (or a re-ranker) to judge whether TM or SMT gives a better translation sentence, and then delivers the better one to the post-editor (He et al., 2010a; He et al., 2010b; Dara et al., 2013). Since the outputs of SMT and TM are not merged but only re-ranked, the possible improvement resulted from those approaches is quite limited.

The second category incorporates TM matched parts into the SMT input sentence in a pipelined manner (Koehn and Senellart, 2010; Zhechev and van Genabith, 2010; He et al., 2011; Ma et al., 2011). These approaches usually translate the sentence in two stages: (1) first determine whether the

extracted TM sentence pair should be adopted or not, and then merge the relevant translations of matched parts into the input sentence; (2) then force the SMT system to only translate those unmatched parts at decoding. There are three drawbacks for this kind of pipeline approaches (Wang et al., 2013). Firstly, whether those matched parts should be adopted or not is determined at the sentence level. Secondly, they select only one TM target phrase before decoding. Thirdly, they do not utilize the SMT probabilistic information for the matched parts.

The third category mainly adds the longest matched TM phrase pairs into the SMT phrase table (Bi ̧ici and Dymetman, 2008; Simard and Isabelle, 2009), and associates them with a fixed large probability value to favor the TM target phrase. However, they only add one aligned target phrase for each matched source phrase and did not distinguish the original and the newly-added phrase-pairs.

The last category incorporates the associated TM information of each source phrase into the SMT during decoding (Wang et al., 2013). This category can avoid the drawbacks of the pipeline approaches, and thus achieves superior results when the TM database and the SMT training set are the same. However, they only refer to the TM information and do not regard the TM phrase-pairs as candidates during decoding. Therefore, the superiority of this approach disappears when the TM database and the SMT training set are different, because many TM phrase-pairs cannot be found in the original SMT phrase table in this case.

Our approach combines the strength of both the third and the last categories. During decoding, the associated TM information is referred to re-score the SMT candidates. At the same time, all matched TM phrase-pairs are dynamically merged into the phrase table. Moreover, this is the first unified framework for integrating TM into SMT at decoding when the TM database and the SMT training set are different. Although some previous works of the second and third categories can be also applied when the TM database and the SMT training set are different, they did not explicitly focus on and test this case.

Last, since the example-based machine translation (EBMT, [Nagao, 1984]) is similar to that of using TM, some approaches (Watanabe and Sumita, 2003; Smith and Clark, 2009; Dandapat et al., 2011; 2012; Phillips, 2011) also combined EBMT with SMT. It would be interesting to compare our approaches with theirs in the future.

6 Conclusion

Combining TM and SMT can greatly improve the translation performance and reduce human post-editing effort. In comparison with those previous approaches, our work makes the following contributions:

(1) Dynamically merge the matched TM phrase-pairs into the SMT phrase table to meet the real application;
(2) Propose an improved integrated model to distinguish the original SMT phrase-pairs from the newly-added ones extracted from TM;
(3) Adopt a simple but effective TM adaptation method to favor the consistent translation in cross-domain test.

This is the first work adopting a unified framework to integrate the TM information into the SMT model during decoding when the TM database and the SMT training set are different. On the TM database which consists of Chinese–English computer technical documents, our experiments have shown that merging the TM phrase-pairs achieves significant improvements when the fuzzy match score is above 0.5. Furthermore, the proposed approaches are significantly better than either the SMT or the TM systems for both the in-domain and the cross-domain tests. Last, the proposed approaches outperform previous works significantly in all test conditions.

Acknowledgements

This research work was partially funded by the Natural Science Foundation of China under Grant No. 61333018, the Hi-Tech Research and Development Program ("863" Program) of China under Grant No. 2012AA011101, the Key Project of Knowledge Innovation Program of Chinese Academy of Sciences under Grant No. KGZD-EW-501, and Toshiba (China) R&D Center.

Reference

Ergun Bi ̧ici and Marc Dymetman. 2008. Dynamic translation memory: using statistical machine translation to improve translation memory fuzzy matches. In *Proceedings of the 9th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2008)*, pages 454–465.

Nicola Cancedda. 2012. Private Access to Phrase Tables for Statistical Machine Translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 23–27.

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. *Technical Report TR-10-98*, Harvard University Center for Research in Computing Technology.

Chiang, David. 2005. A hierarchical phrase-based model for statistical machine translation, In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL)*, pages 263–270.

Sandipan Dandapat, Sara Morrissey, Andy Way, and Mikel L Forcada. 2011. Using example-based MT to support statistical MT when translating homogeneous data in resource-poor settings, In *Proceedings of the 15th Annual Meeting of the European Association for Machine Translation (EAMT 2011)*, pages 201–208.

Sandipan Dandapat, Sara Morrissey, Andy Way, and Joseph Van Genabith. 2012. Combining EBMT, SMT, TM and IR technologies for quality and scale, In *Proceedings of the Joint Workshop on Exploiting Synergies between Information Retrieval and Machine Translation (ESIRMT) and Hybrid Approaches to Machine Translation (HyTra)*, pages 48–58.

Aswarth Dara, Sandipan Dandapat, Declan Groves, and Josef van Genabith. TMTprime: a recommender system for MT and TM integration. In *Proceedings of the NAACL HLT 2013 Demonstration Session*, pages 10–13.

Yifan He, Yanjun Ma, Josef van Genabith and Andy Way, 2010a. Bridging SMT and TM with translation recommendation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 622–630.

Yifan He, Yanjun Ma, Andy Way, and Josef Van Genabith. 2010b. Integrating N-best SMT outputs into a TM system, In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 374–382.

Yifan He, Yanjun Ma, Andy Way and Josef van Genabith. 2011. Rich linguistic features for translation memory-inspired consistent translation. In *Proceedings of the Thirteenth Machine Translation Summit*, pages 456–463.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 181–184.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 388–395, Barcelona, Spain.

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

Philipp Koehn, Franz Josef Och and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 48–54.

Philipp Koehn and Jean Senellart. 2010. Convergence of translation memory and statistical machine translation. In *AMTA Workshop on MT Research and the Translation Industry*, pages 21–31.

Liu, Qun and Hongmei Zhao. 2009. Report on CWMT2009 MT Translation Evaluation. In *Proceedings of the 5th China Workshop on Machine Translation (CWMT2009)*, pages 1–31, Nanjing, China.

Yanjun Ma, Yifan He, Andy Way and Josef van Genabith. 2011. Consistent translation using dis-criminative learning: a translation memory-inspired approach. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1239–1248, Portland, Oregon.

Makoto Nagao, 1984. A framework of a mechanical translation between Japanese and English by anal-ogy principle. In: Banerji, Alick Elithorn and Ran-an (ed). *Artifiical and Human Intelligence: Edited Review Papers* Presented at the International NATO Symposium on Artificial and Human Intelligence. North-Holland, Amsterdam, 173–180.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 160–167.

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

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

Aaron B. Phillips, 2011. Cunei: open-source machine translation with relevance-based models of each translation instance. *Machine Translation*, 25 (2). pages 166-177.

Michel Simard and Pierre Isabelle. 2009. Phrase-based machine translation in a computer-assisted translation environment. In *Proceedings of the Twelfth Machine Translation Summit (MT Summit XII)*, pages 120–127.

James Smith and Stephen Clark. 2009. EBMT for SMT: a new EBMT-SMT hybrid. In *Proceedings of the 3rd International Workshop on Example-Based Machine Translation (EBMT'09)*, pages 3–10, Dublin, Ireland.

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 Association for Ma-chine Translation in the Americas (AMTA-2006)*, pages 223–231.

Andreas Stolcke. 2002. SRILM-an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, pages 311–318.

Taro Watanabe, Eiichiro Sumita. 2003. Example-based decoding for statistical machine translation, In *Proceeding of Machine Translation Summit IX*, pages 410–417.

Kun Wang, Chengqing Zong and Keh-Yih Su, 2013. Integrating translation memory into phrase-based machine translation during decoding. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 11–21.

Ventsislav Zhechev and Josef van Genabith. 2010. Seeding statistical machine translation with translation memory output through tree-based structural alignment. In *Proceedings of the 4th Workshop on Syntax and Structure in Statistical Translation*, pages 43–51.

Xiaoning Zhu, Zhongjun He, Hua Wu, Haifeng Wang, Conghui Zhu, and Tiejun Zhao. 2013. Improving pivot-based statistical machine translation using random walk. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 524–534.