Available online at www.sciencedirect.com

ScienceDirect

Computer Speech & Language 54 (2019) 1–31

www.elsevier.com/locate/csl

A unified framework and models for integrating translation memory into phrase-based statistical machine translation ☆

Yang Liu*[,a], Kun Wang[a], Chengqing Zong[a], Keh-Yih Su[b]

[a] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[b] Institute of Information Science, Academia Sinica, Taipei, Taiwan

Received 3 October 2017; received in revised form 5 March 2018; accepted 16 September 2018
Available online 23 October 2018

Abstract

Since statistical machine translation (SMT) and translation memory (TM) complement each other in TM matched and unmatched regions, a unified framework for integrating TM into phrase-based SMT is proposed in this paper. Unlike previous two-stage pipeline approaches, which directly merge TM results into the input sentences and subsequently let the SMT only translates those unmatched regions, the proposed framework refers to the corresponding TM information associated with each phrase at the SMT decoding. Under this unified framework, several integrated models are proposed to incorporate different types of information extracted from TM to guide the SMT decoding. We thus let SMT implicitly and indirectly utilize global context with a local dependency model. Furthermore, the SMT phrase table is dynamically enhanced with TM phrase pairs when the TM database and the SMT training set are different.

On a Chinese−English TM database, our experiments show that the proposed Model-I significantly improves over both SMT and TM when the SMT training set is also adopted as the TM database and when the fuzzy match score is over 0.4 (overall 3.5 BLEU points improvement and 2.6 TER points reduction). In addition, the proposed Model-II is significantly better than the TM and the SMT systems when the SMT training set and the TM database are different. Furthermore, the proposed Model-III outperforms both the TM and the SMT systems even when the SMT training set and the TM database are from different domains. Additionally, the proposed Model-IV further achieves significant improvements with the help of Top-N TM sentence pairs. Lastly, all our models significantly outperform those state-of-the-art approaches under all test conditions.

Keywords: Phrase-based machine translation; Translation memory

1. Introduction

Statistical machine translation (SMT), especially the phrase-based approach (Koehn et al., 2003), advances very fast since the beginning of this century (Yamada and Knight, 2001; Och and Ney, 2002; Chiang, 2005; Huang and Chiang, 2007; Koehn et al., 2007; Watanabe et al., 2007; Chiang et al., 2008; Galley and Manning, 2008; Hopkins

☆ This paper has been recommended for acceptance by Pascale Fung

* Corresponding author.
E-mail addresses: yang.liu2013@nlpr.ia.ac.cn (Y. Liu), kunwang.work@aliyun.com (K. Wang), cqzong@nlpr.ia.ac.cn (C. Zong), kysu@iis.sinica.edu.tw (K.-Y. Su).

https://doi.org/10.1016/j.csl.2018.09.006
0885-2308/

and May, 2011; Cherry, 2013; Galley et al., 2013). For certain language pairs and special applications, SMT output has reached an acceptable level, especially in the domains where abundant parallel corpora are available (He et al., 2010a). An increasing number of commercial corporations, such as Google, Microsoft, and Baidu, have provided free online SMT services. However, SMT is still seldom adopted in the professional translation circle because its outputs are usually poor and far from satisfactory. Specifically, there is no guarantee that an SMT system can produce translations in a consistent manner (Ma et al., 2011b). Therefore, to reach the required quality, considerable manual post-editing is usually essential (Barrachina et al., 2009), and it makes adopting SMT uneconomical.

In contrast, translation memory (TM), which uses the most similar translation sentence (usually above a certain fuzzy match threshold) in the database as the reference for post-editing, has been widely adopted in the professional translation community for many years (Lagoudaki, 2006). Usually, TM is embedded within a Computer Assisted Translation (CAT) toolkit (such as Trados,[1] DéjàVu[2] and Wordfast)[3] used by human translators. The aim of TM is to avoid duplication work and to improve the working efficiency. TM is very useful for repetitive material, such as updated product manuals, and can provide high-quality and consistent translations when the similarity of the fuzzy match is high. Therefore, professional translators trust TM much more than SMT. However, high-similarity fuzzy matches are available only when the material is quite repetitive. Additionally, as it is only translated from a similar sentence, it cannot be directly used as the proper translation without editing.

By comparing SMT with TM under different fuzzy match ratios (Koehn and Senellart, 2010), it shows that TM outperforms SMT in those high-similarity cases (i.e., when the fuzzy match score is higher than 0.8 in our task), whereas SMT exceeds TM in those low-similarity cases. Further, inspection reveals that TM provides more reliable results for those matched sub-segments. This is because the current SMT system only utilizes local context (due to the incapability of handling long distance dependence), whereas TM are generated by human according to the global context. However, for those unmatched sub-segments, SMT gives more reliable results because it does translate the real input. Since TM and SMT complement each other in those matched and unmatched sub-segments, the output quality is expected to be significantly higher (resulting in a substantially reduced manual postediting effort) if they can be combined to supplement each other. Furthermore, when the similarity is high, more TM matched target phrases can be referred by the SMT; SMT thus will be able to indirectly enjoy the global context, although its original model can only handle local context.

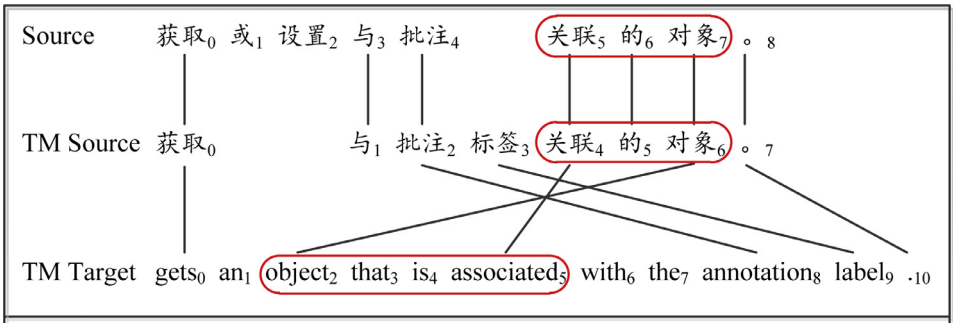
Since SMT and TM have been separately developed within different communities, only a few studies (Biçici and Dymetman, 2008; Simard and Isabelle, 2009; He et al., 2010a; 2010b; Koehn and Senellart, 2010; Zhechev and Genabith, 2010; Ma et al., 2011a; Ma et al., 2011b; Dara et al., 2013) have been conducted to combine them. According to the method of combination, those previous studies can be classified into three categories: (1) simply selecting the better translation sentence among SMT and TM, (2) incorporating TM-matched subsegments into SMT and only translating those unmatched parts, and (3) only enhancing the SMT phrase table with new TM phrase pairs.

The first category uses a classifier (or a re-ranker) to judge whether TM or SMT gives a better translation sentence and subsequently delivers the better one to the post-editor (He et al., 2010a; 2010b; Dara et al., 2013). Because the SMT and the TM outputs are not merged but only re-ranked for the post-editors, the possible improvements resulting from these approaches are quite limited.

The second category incorporates TM matched sub-segments into SMT in a pipelined manner (Koehn and Senellart, 2010; Zhechev and Genabith, 2010; Ma et al., 2011a; Ma et al., 2011b). All these pipeline approaches translate the sentences in two stages. They first determine whether the extracted TM sentence pair should be adopted or not. Most of them (Koehn and Senellart, 2010; Zhechev and Genabith, 2010) use a pre-specified fuzzy match score as the threshold. Afterwards, they merge the relevant translations of matched sub-segments into the input sentence and then force the SMT system to only translate those unmatched sub-segments at decoding.

Nonetheless, there are three drawbacks for those pipeline approaches. First, they all determine whether those matched sub-segments should be adopted or not at the sentence level. Specifically, matched sub-segments are either all adopted or all abandoned regardless of their individual quality; however, not every TM matched sub-segment should be adopted or abandoned. Second, as several TM target phrases might be available for one given TM source phrase due to word insertions, an incorrect selection made in the merging stage cannot be remedied in the following

[1] http://www.translationzone.com/trados.html
[2] http://www.atril.com/
[3] http://www.wordfast.com/



TM Source Phrase	关联$_4$ 的$_5$ 对象$_6$
TM Target Phrase 1	object$_2$ that$_3$ is$_4$ associated$_5$
TM Target Phrase 2	an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$
TM Target Phrase 3	object$_2$ that$_3$ is$_4$ associated$_5$ with$_6$
TM Target Phrase 4	object$_2$ that$_3$ is$_4$ associated$_5$ with$_6$ the$_7$
TM Target Phrase 5	an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$ with$_6$
TM Target Phrase 6	an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$ with$_6$ the$_7$

Fig. 1. An example of multiple phrase-mappings.

translation stage. For example, there are six possible corresponding TM target phrases for the given TM source phrase "关联$_4$ 的$_5$ 对象$_6$" (as shown in Fig. 1), such as "*object$_2$ that$_3$ is$_4$ associated$_5$*" and "*an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$ with$_6$*", etc. It is difficult to determine which one should be adopted in the merging stage. Third, the pipeline approach does not utilize SMT probabilistic information in deciding whether a matched TM phrase should be adopted or not and which TM target phrase should be selected when we have multiple candidates. Therefore, the approaches of this category still cannot produce satisfactory results.

The last category mainly adds the longest matched TM phrase pair into the SMT phrase table (Biçici and Dymetman, 2008; Smith and Clark, 2009) and associates them with a fixed large probability value to favor the TM target phrase during SMT decoding. Since only one aligned target phrase will be added for each matched source phrase, they share most of the drawbacks of the pipeline approaches mentioned above and merely achieve similar performance.

To avoid the above mentioned drawbacks of the pipeline approaches (which are mainly due to making a hard decision *before* decoding), we propose a unified framework to completely make use of TM information *during* decoding. First, for each matched TM source phrase, we keep all its possible corresponding target phrases instead of keeping only one of them. Furthermore, we dynamically merge those TM matched phrase pairs into the SMT phrase table for each sentence when the TM database and the SMT training set are different. Several integrated models are then proposed to jointly consider all corresponding TM target phrases and the given SMT target phrase candidate during decoding. Therefore, under this unified framework, the proposed integrated models combine SMT and TM at a *deep* level, which allows SMT to utilize global context with a local dependency model. This is a significant departure from previous approaches which directly plugged the TM result into the final output at the *surface* level.

To verify the superiority of the proposed approach, various experiments have been conducted on a Chinese-English computer technical documents TM database. When the TM database and the SMT training set are the same, our experiments show that the proposed Model-I significantly improves the translation quality over both the phrase-based SMT system and the TM system when the fuzzy match score is above 0.4. Compared to the SMT system, it achieves a 3.5 BLEU points improvement and 2.6 TER points reduction overall. In addition, when the TM database and the SMT training set are different but from the same domain, our experiments show that dynamically merging TM phrase pairs into the SMT phrase table considerably improves the translation quality, and Model-II is significantly better than both the SMT and the TM systems by additionally distinguishing TM phrase pairs from the

original SMT phrase pairs. Furthermore, the proposed Model-III (which also checks if there is an exactly matched TM target phrase candidate for the given SMT target phrase) significantly outperforms both the TM and the SMT systems even when the TM database and the SMT training set are from different domains. And its adapted version additionally considers the domain-mismatch problem to boost the performance further. Afterwards, with the help of the Top-N relevant TM sentence pairs, it is extended to simultaneously utilize multiple TM sentence pairs to increase matched regions (and is called Model-IV). In summary, all proposed models significantly outperform those state-of-the-art approaches in all test conditions.

In comparison with our simplified version (i.e., Model-I) previously published in the ACL and the COLING conferences (Wang et al., 2013; 2014), this article re-formulates the problem for given multiple relevant TM sentence pairs and then re-derives the formulation (i.e., Model-IV), whereas the original versions only utilized the most similar TM sentence pair. Also, two additional models (i.e., Model-II and III) are proposed in this article to handle more existing scenarios, and their related experiments are added.

The remainder of this paper is organized as follows: Section 2 presents the proposed unified framework and its associated algorithms. Section 3 introduces the adopted data sets and four different baseline systems. Afterwards, centering on three different operation scenarios, various models are proposed and their corresponding experiments are introduced in Section 4. Section 5 then reviews related work. Finally, conclusions and future work are given in Section 6.

2. The unified framework

In order to better understand our work, the proposed framework is first illustrated in Section 2.1. Afterwards, two related procedures for implementing it are described in Sections 2.2 and 2.3, respectively.

2.1. Problem formulation

To extract closely matched TM sentence pairs from the database, the *Fuzzy Match Score* (FMS) is commonly adopted to measure the overall similarity between the input sentence and a given TM source sentence *tm_s*. In this paper, the FMS is defined as (Sikes, 2007):

$$FMS(s, tm\_s) = 1 - \frac{Levenshtein(s, tm\_s)}{max(|s|, |tm\_s|)} \tag{1}$$

Where Levenshtein(s, *tm_s*) is the word-based *Levenshtein Distance* between s and *tm_s* (also known as the *Edit Distance*). This distance is the number of *deletions, insertions*, and *substitutions* required to transform s into *tm_s*, and FMS is a length-normalized variant of the Levenshtein Distance. For example, the FMS between the source sentence and the TM source sentence in Fig. 1 is 0.667 (two deletions and one insertion for nine words).

With the extracted most similar TM sentence pair, the translation problem can be formulated as:

$$\widehat{t} = argmax_t\, P(t|s, [tm\_s, tm\_t, tm\_f, s\_a, tm\_a]) \tag{2}$$

Where s denotes the given source sentence; t is a corresponding target sentence; $\widehat{t}$ is the final translation; [*tm_s*, *tm_t*, *tm_f*, *s_a*, *tm_a*] denotes the associated information of the most similar TM sentence pair extracted from the TM database; *tm_s* and *tm_t* are the corresponding source and target sentences, respectively; *tm_f* denotes its corresponding *FMS*, as defined above; *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*.

As we would like to integrate TM into the phrase-based SMT framework, the sentences are first converted into their associated phrase sequences (see Appendix A for the derivation). And the problem is reformulated as:

$$\widehat{t} \equiv argmax_t\, P(\overline{t}_1^K | \overline{s}_{a(1)}^{a(K)}, [tm\_s, tm\_t, tm\_f, s\_a, tm\_a]) \tag{3}$$

In the above formulation, it is assumed that there are a total of K phrase-pairs without phrase insertions (if a source phrase is deleted, then its corresponding target phrase would be "ϕ"). Also, $\overline{s}_{a(1)}^{a(K)}$ and $\overline{t}_1^K$ denote the associated source phrase sequence and the target phrase sequence, respectively. With the help of the corresponding editing operations *s_a* (via monolingual word alignment), for any given source phrase $\overline{s}_{a(k)}$, we can find its corresponding TM source phrase $tm\_\overline{s}_{a(k)}$ from *tm_s*. Additionally, we can extract the corresponding TM target phrase according to

the word alignment between *tm_s* and *tm_t* (see the next *Aligning TM Phrases* section). However, as illustrated in Fig. 1, we might have multiple TM target-phrases $tm\_\bar{t}_{a(k)}$ for the same TM source phrase $tm\_\bar{s}_{a(k)}$ due to word insertions. We will consider all those possible TM target phrases during decoding. By introducing various TM target phrases $tm\_\bar{t}_{a(k)}$ and by quantizing the fuzzy match score tm_f into its corresponding interval (to be specified in Section 3.1), the probability factor in Eq. (3) can be further derived as follow:

$$
\begin{aligned}
&P(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}, [tm\_s, tm\_t, tm\_f, s\_a, tm\_a]) \\
&= \sum_{tm\_\bar{t}_{a(1)}^{a(K)}} P\left(\bar{t}_1^K, tm\_\bar{t}_{a(1)}^{a(K)} \middle| \bar{s}_{a(1)}^{a(K)}, tm\_\bar{s}_{a(1)}^{a(K)}, tm\_t, z\right) \approx max_{tm\_\bar{t}_{a(1)}^{a(K)}} \left\{ P\left(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}\right) \right. \\
&\left. \times P\left(tm\_\bar{t}_{a(1)}^{a(K)}|\bar{s}_{a(1)}^{a(K)}, tm\_\bar{s}_{a(K)}^{a(1)}, tm\_t, z\right) \right\} \approx \max_{tm\_\bar{t}_{a(1)}^{a(K)}} \left\{ P\left(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}\right) \times P\left(M_1^K|L_1^K, tm\_t, z\right) \right\} \\
&\approx P\left(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}\right) \times \prod_{k=1}^{K} \max_{tm\_t_{a(k)}} P(M_k|L_k, z)/C_k
\end{aligned} \tag{4}
$$

In the above derivation, the first factor $P\left(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}\right)$ is simply the typical phrase-based SMT model and the second factor $P(M_k|L_k, z)$ is the information derived from the given TM sentence pair. Therefore, we can still keep the original phrase-based SMT model and only pay attention to how to extract useful information from the associated TM sentence pair to guide the new SMT decoding, which is the motivation of this derivation.

Furthermore, $P(M_k|L_k, z)$ in Eq. (4) will be the meta/core probability factor shared among various models; of which M_k is the corresponding TM target phrase matching status for the current target candidate $\bar{t}_k$, which is a vector consisting of various indicators (e.g., "*Target Phrase Content Matching Status*", etc., to be defined in Section 4) and reflects the quality of the given candidate, and L_k is the linking status vector of $\bar{s}_{a(k)}$ (the aligned source phrase, within $\bar{s}_1^K$, 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). M_k and L_k are just meta symbols shared by various models, and will be instantiated and specified for each model in Section 4. Finally, $C_k = \sum_{tm\_\bar{t}_{a(k)}} P(M_k|L_k, z)$ is a normalization value,[4] which will be ignored during decoding.

In the second line of Eq. (4), we incorporate all possible combinations of TM target phrases. We then only select the best one in the third line for simplicity. Afterwards, we introduce the source linking status L_k and the target matching status M_k. Since we might have several possible TM target phrases $tm\_\bar{t}_{a(k)}$, the one with the maximum score will be adopted for reference at the last line. Therefore, $P(M_k|L_k, z)$ is designed to assign preference to each translation candidate $\bar{t}_k$ according to its target matching status M_k and is used to favour the SMT target-phrase candidate that is more similar (in both content and position) to those matched TM target phrases.

The detailed steps for this unified framework would thus be as follows: for each given source phrase $\bar{s}_{a(k)}$, we first check its TM linking status L_k on the source side. Afterwards, for each possible SMT target phrase $\bar{t}_k$ associated with $\bar{s}_{a(k)}$, we obtain all its aligned TM target phrases $tm\_\bar{t}_{a(k)}$ on the target side. Next, for each $tm\_\bar{t}_{a(k)}$, we calculate its corresponding $P(M_k|L_k, z)$. Finally, the highest $P(M_k|L_k, z)$ (among various $tm\_\bar{t}_{a(k)}$) is selected as the corresponding TM score for $\bar{t}_k$ to guide the SMT decoding. Fig. 2 shows the corresponding decoding flow.

2.2. Aligning TM phrases

To incorporate TM information under the unified framework, we need to solve the following two problems in advance: (1) identifying the corresponding TM source phrase $tm\_\bar{s}_{a(k)}$ for a given source phrase $\bar{s}_{a(k)}$, and (2) extracting the corresponding TM target phrases $tm\_\bar{t}_{a(k)}$ for $tm\_\bar{s}_{a(k)}$.

For the first problem, we first obtain the corresponding editing operations s_a while calculating the Levenshtein Distance between the source sentence s and the TM source sentence tm_s via dynamic programming. Afterwards, the corresponding $tm\_\bar{s}_{a(k)}$ can be directly obtained by locating those associated TM source words with the "*match*" operation. For example, the editing operations between the source sentences in Fig. 1 are "*m d d m m i m m m m*" ("m", "d" and "i" denote "match", "deletion" and "insertion", respectively). Since each "*match*" denotes a link

[4] The summation will be taken over various $tm\_\bar{t}_{a(k)}$ that are associated with tm_t.

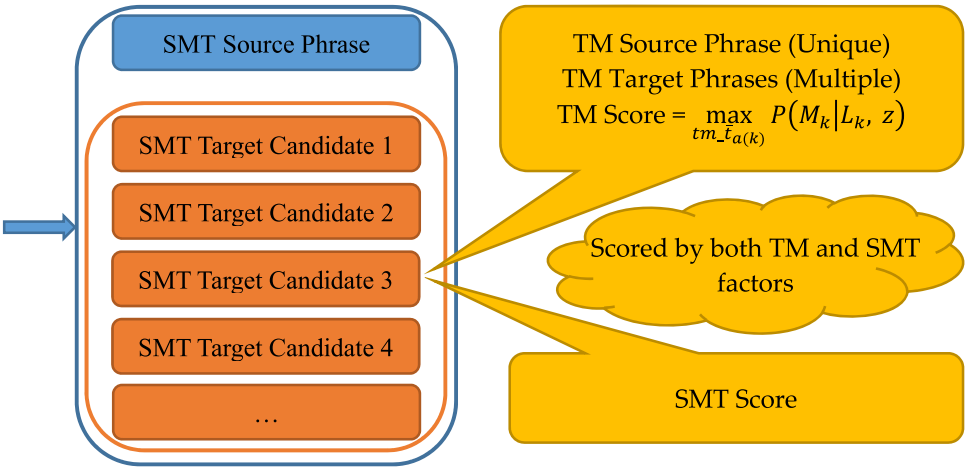


Fig. 2. Decoding flow of the proposed unified framework.

between two words, we can find the following Input-TM source phrase mappings: 【关联$_5$ 的$_6$ 对象$_7$ → 关联$_4$ 的$_5$ 对象$_6$】, 【批注$_4$ 关联$_5$ → 批注$_2$ 标签$_3$ 关联$_4$】, 【或$_1$ 设置$_2$ → NA】, etc. Note that both "或$_1$" and "设置$_2$" are deleted words; thus, there is no corresponding TM source phrase for "或$_1$ 设置$_2$" ("NA" is thus used to denote this case).

For the second problem, $tm\_\bar{t}_{a(k)}$ can be obtained from $tm\_\bar{s}_{a(k)}$ via their bilingual word alignment links. For example, for the given TM source phrase "关联$_4$ 的$_5$ 对象$_6$", the corresponding TM target phrase according to the bilingual word alignment is "$object_2$ $that_3$ is_4 $associated_5$". Because there are three word insertions at the boundaries (one at the left boundary and two at the right boundary), we can obtain five additional corresponding target phrases (shown in Fig. 1) according to the phrase extraction heuristics.

2.3. Merging TM phrase pairs

The phrase-based SMT needs a phrase table to operate. Since the same phrase extraction heuristics are adopted to extract both the SMT and TM phrase pairs, the SMT phrase table will cover all the continuous TM phrase pairs within the phrase length limit when the TM database and the SMT training set are the same. However, this would not be true when the TM database and the SMT training set are different. Therefore, the SMT phrase table could be further enhanced with the 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 the TM source phrase exists in the original SMT phrase table, but its corresponding target phrase does not; and (3) even the 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 phrases from the second and the third categories should be dynamically added to the SMT phrase table for each input sentence. To distinguish those newly added phrase pairs from the original SMT phrase pairs, we will use eight additional feature weights λ_m for the translation probability (lexicon and phrase transfer in both directions) and two more feature weights for the phrase penalty (details will be specified later in Section 4.2.2).

The above approach is inspired by the work of Biçici and Dymetman (2008). However, there are three differences between our approach and theirs. First, we add all the matched TM phrase pairs (including all associated sub-phrase pairs),[5] whereas they only added the longest matched phrase pairs. Second, we add all the possible TM target phrase pairs for a given TM source phrase, whereas they only extracted one TM target phrase regardless of the existence of multiple TM target candidates. Finally, we use different feature weights to distinguish those newly added TM phrase pairs from the original SMT phrase pairs, whereas they treated them equally.

[5] One phrase-pair may contain various sub-phrase pairs.

Table 1
Corpus statistics of the TM dataset.

	#Sentences	#Chn. words	#Chn. VOC.	#Eng. words	#Eng. VOC.
TM dataset	267,051	3,700,749	49,895	3,703,867	52,260
Develop	2569	38,585	3287	38,329	3993
Test	2576	38,648	3460	38,510	4046

2.4. Cross-fold translation

To estimate the associated probabilities of the proposed models, the corresponding phrase segmentations for bilingual sentences are required. To simulate the test set situation, cross-fold translation is thus used in advance to obtain the desired phrase segmentations. We first extract 95% of the specified SMT training set as a new training corpus to train an SMT model. Afterwards, we generate the corresponding phrase segmentations for the remaining 5% of the bilingual sentences using *forced decoding* (Li et al., 2000; Zollmann et al., 2008; Auli et al., 2009; Wisniewski et al., 2010), which searches the best phrase segmentation for the specified output. Having repeated the above steps 20 times,[6] we obtain the corresponding phrase segmentations for the whole SMT training dataset (which will then be used to train the integrated models).

Due to OOV and insertion words, not all given source sentences can result in the desired results through forced decoding. Fortunately, most of the training sentences can result in the corresponding results in our work. For example, in the experiments where the TM database is also adopted as the SMT training set (Section 4.1.2), 71.7% of the training sentence pairs can result in the desired target results. The remaining 28.3% of the sentences pairs are thus not adopted for generating $P(M_k|L_k, z)$ training samples (but they are still adopted for training the SMT model). Furthermore, more than 90% of the obtained source phrases are observed to be less than five words, which explains why five different quantization levels are adopted for the*Source Phrase Length* (SPL) in Section 4.1.1.

3. Data sets and baselines

Since various data sets and baselines will be shared among different proposed models and experiments, their descriptions are first given here. Section 3.1 describes the data sets we adopt, and the description of baselines is given at Section 3.2.

3.1. Adopted data sets

To evaluate the performance of the following proposed models, we conduct experiments in Chinese−English language pair. Our TM dataset consists of Chinese−English translation computer domain technical document sentence pairs and contains approximately 267k sentence pairs. The average length of the Chinese sentences is 13.85 words, and that of the English sentences is 13.86 words. All the experiments are conducted around this TM dataset. To compare the performances under various conditions, the same development and test sets will be shared by all different experiments, which are randomly selected from the above TM dataset first. Subsequently, the remaining sentence pairs are used as either the SMT training set or the TM database under different scenarios. The detailed corpus statistics are shown in Table 1. Because the associated SMT training set and the TM database will vary under different experimental configurations, they will be specified later in their related sub-sections.

3.2. Adopted baselines and evaluation metrics

The baselines we use for comparison include: TM, SMT, Koehn-10, and Ma-11-U four systems (to be specified below). Koehn-10 and Ma-11-U are selected because they report superior performances in the literature. The translation performance is measured using case-insensitive BLEU-4 (Papineni et al., 2002) and TER scores (Snover et al., 2006).

[6] This training process only took approximately 10 h on our Ubuntu server (Intel 4-core Xeon 3.47 GHz, 132 GB of RAM).

Statistical significance tests are conducted with the re-sampling approach (1000 times) at a 95% confidence level (Koehn, 2004). The detail of baseline systems is as follows:

- *Translation memory system* (TM): This TM system keeps all the aligned-sentence-pairs that the human translator has created in a TM *database*. When the human translator works on a new source document on a TM system, for each source-sentence in the new document, TM system compares it with all the source-sentences kept in the TM database. For each aligned-sentence-pair, the system will generate a corresponding Fuzzy Match Score between two source-sentences (one is the input sentence and one is that kept in the TM database). The system will locate the aligned-sentence-pair which possesses the highest FMS with the input sentence, and then fetch the associated target-sentence of that aligned-sentence-pair as the translation memory. In this case, we use the word-based fuzzy match score (see Section 2.1) as the similarity measure.
- **Statistical machine translation (SMT)**: For the phrase-based translation model, we adopt the Moses toolkit (Koehn et al., 2007). The following typical features are adopted: the phrase translation model scores, language model probability, distance-based reordering score, lexicalized reordering model scores and word penalty. The system configuration is given as follows: GIZA++ (Och, 2003) is used to obtain the bidirectional word alignments. Subsequently, "intersection" refinement (Koehn et al., 2003) is adopted to extract phrase pairs. We use the SRI Language Model toolkit (Stolcke et al., 2002) to train a 5-gram model using modified Kneser−Ney smoothing (Kneser and Ney, 1995; Chen and Goodman, 1996) on the target-side (English) training corpus. All the feature weights and the weight for each probability factor (for example, three factors for Model-I) are tuned on the development set with minimum-error-rate training procedure (MERT) (Och, 2003). The maximum phrase length is set to seven in our experiments.
- *Koehn-10:* We re-implement the previous work from Koehn and Senellart (2010), which is an XML-Markup approach. The main idea is as follows: If the extracted most similar TM sentence pair is determined to be adopted, they first detect the matched and unmatched parts and subsequently keep the matched parts with their corresponding translations; subsequently, the SMT system only translates those unmatched parts. Therefore, it is also a pipelined approach. In the case of Koehn-10, they used the fuzzy match score to decide whether the TM sentence should be adopted or not.
- *Ma-11-U:* Ma et al. (2011a) also proposed an XML-Markup approach. Instead of using the above fuzzy match score (adopted in Koehn-10), they adopted a discriminative classifier to determine whether the TM sentence should be adopted. Ma-11-U is obtained by only re-implementing their XML-Markup method used in Ma et al. (2011b) and Ma et al. (2011a), but not their discriminative learning method. This is because the features adopted in their discriminative learning method are complicated and difficult to re-implement. Since the oracle classification is assumed in Ma-11-U, it is thus an upper bound for Ma et al. (2011b). Besides, because Ma et al. (2011a) only added additional linguistic features to the classification model, this upper bound is also applied for Ma et al. (2011b).

Although both Koehn-10 and Ma-11-U adopt the XML-Markup approach, there is a subtle difference between them. In Koehn and Senellart (2010), for each *unmatched* phrase in the TM source sentence, they replaced its corresponding translation in the *TM target sentence* by the corresponding source phrase in the input sentence and subsequently mark the substitution part. After replacing the corresponding translations of all unmatched source phrases in the TM target sentence, an XML input sentence (with mixed TM target phrases and marked input source phrases) is obtained. The SMT decoder then only translates the unmatched/marked source phrases and obtains the desired results. Therefore, the *inserted* parts in the TM target sentence are automatically *included*. In contrast, Ma et al. (2011a) replaced each matched source phrase in the given source sentence with the corresponding TM target phrase. Therefore, the *inserted* parts in the TM target sentence were *excluded*. Fig. 3 presents the difference between the two XML-Markup methods.[7] It can be observed that the inserted parts "the" and "on your" are automatically included in Koehn-10 but not in Ma et al. (2011a). The drawbacks of these approaches have been mentioned in Section 1 and will not be repeated here.

[7] To show the difference more clear, the word alignments in Fig. 3 are not obtained from GIZA++ but are done by hand.

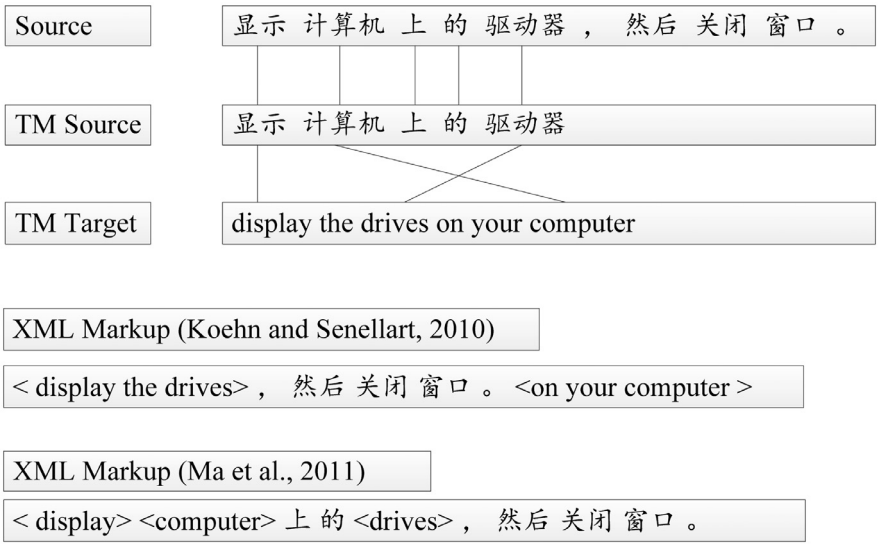


Fig. 3. The difference between two XML-Markup methods.

4. Proposed models and results

Under the unified framework mentioned above (Eq. (4)), we propose four different models which incorporate various types of TM information under three different scenarios. Section 4.1 discusses the first scenario in which the TM database is also adopted as the SMT training set. Afterwards, the second scenario, in which a different SMT training set from the same domain is adopted, is discussed in Section 4.2. Finally, Section 4.3 discusses the last scenario in which a different cross-domain SMT training set is adopted. Along this scenario sequence, each later one gets closer to the real application. Furthermore, we use the Factored Language Model toolkit (Koehn et al., 2007) to estimate the probabilities of various models with Witten−Bell smoothing (Bell et al., 1990; Witten and Bell, 1991) and the Back-off[8] method.

4.1. Adopting the TM database as the SMT training set

This scenario is artificial as TM database is also the SMT training set. Since they are the same, the following proposed Model-I actually can be regarded as a *new* phrase-based SMT model which can implicitly utilize the global context.

4.1.1. Proposed Model-I

In this model, the meta/core factor $P(M_k|L_k, z)$ in Eq. (4) is instantiated with various features shown in the following equation:

$$\begin{aligned} & P(M_k|L_k, z) \\ & \triangleq P([RPM, TCM, CRL]_k|[SCM, NLN, CSS, SPL, SEP]_k, z) \\ & \approx \left\{ \begin{array}{l} P(TCM_k|SCM_k, NLN_k, CRL_k, SPL_k, SEP_k, z) \\ \quad \times P(CRL_k|CSS_k, SCM_k, NLN_k, SEP_k, z) \\ \quad\quad \times P(RPM_k|TCM_k, SCM_k, NLN_k, z) \end{array} \right\} \end{aligned} \tag{5}$$

[8] Each back-off sequence is ordered by the importance of each feature. The sequence is specified in each probability factor according to the following convention: for $P(X|F_1, F_2, F_3, z)$, the back-off sequence would be F_3, F_2 and then F_1. (Please note that FMZ interval index "z" is always kept.)

- *TM fuzzy match interval index* (z): The fuzzy match score $FMS(s, tm\_s)$ between the source sentence and the TM source sentence tm_s indicates the reliability of the given TM sentence. The higher the fuzzy match score is, the more reliable the TM information is. In our task, this value is equally divided into ten fuzzy match intervals, such as [0.9, 1.0), [0.8, 0.9), etc., and the index z specifies the corresponding interval. For example, the fuzzy match score between the source sentence and TM source sentence in Fig. 1 is 0.667, then $z = [0.6, 0.7)$.
- *Target phrase content matching status* (TCM): The TCM indicates the content matching status between $\overline{t}_k$ and $tm\_\overline{t}_{a(k)}$, and reflects the quality of $\overline{t}_k$. If the similarity between $\overline{t}_k$ and $tm\_\overline{t}_{a(k)}$ is high, then it implies that the given $\overline{t}_k$ is possibly a good candidate. It is a member of {Same, High, Low, NA (Not-Applicable)}, and is specified as follows:

  (1) If $tm\_\overline{t}_{a(k)}$ is not null:

  (a) if $FMS(\overline{t}_k, tm\_\overline{t}_{a(k)}) = 1.0$, then $TCM_k = Same$;
  (b) else if $FMS(\overline{t}_k, tm\_\overline{t}_{a(k)}) > 0.5$, then $TCM_k = High$;
  (c) else, $TCM_k = Low$;

  (2) If $tm\_\overline{t}_{a(k)}$ is null, $TCM_k = NA$.

- *Source phrase content matching status* (SCM): The SCM indicates the content matching status between $\overline{s}_{a(k)}$ and $tm\_\overline{s}_{a(k)}$ and greatly affects the matching status of $\overline{t}_k$ and $tm\_\overline{t}_{a(k)}$. The more similar $\overline{s}_{a(k)}$ is to $tm\_\overline{s}_{a(k)}$, the more likely that $\overline{t}_k$ is similar to $tm\_\overline{t}_{a(k)}$. It is a member of {Same, High, Low, NA} and is specified as follows :

  (1) If $tm\_\overline{s}_{a(k)}$ is not null:

  (a) if $FMS(\overline{s}_{a(k)}, tm\_\overline{s}_{a(k)}) = 1.0$, then $SCM_k = Same$;
  (b) else if $FMS(\overline{s}_{a(k)}, tm\_\overline{s}_{a(k)}) > 0.5$, then $SCM_k = High$;
  (c) else, $SCM_k = Low$;

  (2) If $tm\_\overline{s}_{a(k)}$ is null, $SCM_k = NA$.

  Here, $tm\_\overline{s}_{a(k)}$ is null means that there is no corresponding TM source phrase $tm\_\overline{s}_{a(k)}$ for the given source phrase $\overline{s}_{a(k)}$ in the TM source sentence tm_s. For example, assume that the given source phrase $\overline{s}_{a(k)}$ is "或$_1$ 设置$_2$" in Fig. 1, then the corresponding $tm\_\overline{s}_{a(k)}$ is null because both of the words in $\overline{s}_{a(k)}$ are deleted. Therefore, the current SCM_k is "NA". Take the source phrase $\overline{s}_{a(k)}$ "关联$_5$ 的$_6$ 对象$_7$" in Fig. 1 as another example, since the corresponding $tm\_\overline{s}_{a(k)}$ is "关联$_4$ 的$_5$ 对象$_6$", the current SCM_k is "Same" (as $FMS(\overline{s}_{a(k)}, tm\_\overline{s}_{a(k)}) = 1.0$).
- *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. Therefore, this NLN feature reflects the number of matched neighbors (words) and it is a vector of $< x, y >$. Where "x" denotes the number of matched source neighbors; and "y" denotes how many those neighbors are also linked to target words (not null), which also affects the TM target phrase selection. This feature is a member of $\{ < x, y > : < 2, 2 > , < 2, 1 > , < 2, 0 > , < 1, 1 > , < 1, 0 > , < 0, 0 > \}$. For the source phrase $\overline{s}_{a(k)}$ "关联$_5$ 的$_6$ 对象$_7$" in Fig. 1, the corresponding $tm\_\overline{s}_{a(k)}$ is "关联$_4$ 的$_5$ 对象$_6$". As only their right neighbors "$\circ_8$"and "$\circ_7$" are matched, and "$\circ_7$" is aligned with "$_{.10}$", then NLN_k will be " $< 1, 1 >$ ".
- *Source phrase length* (SPL): Usually, the longer the source phrase is, the more reliable the TM target phrase is (especially for the TM source phrase that is the same with the input source phrase). For example, the corresponding $tm\_\overline{t}_{a(k)}$ for a source phrase with five words would be more reliable than that for a source phrase with only one word. This feature denotes the number of words included within $\overline{s}_{a(k)}$, and is a member of $\{1, 2, 3, 4, \geq 5\}$. Take the source phrase "关联$_5$ 的$_6$ 对象$_7$" in Fig. 1 as an example. In this case, SPL_k will be "3" because it contains three words.
- *Sentence end punctuation indicator* (SEP): The SEP indicates whether the current source phrase $\overline{s}_{a(k)}$ is the punctuation at the end of the sentence, and is a member of {Yes, No}. For example, if the current $\overline{s}_{a(k)}$ is "关联$_5$ 的$_6$ 对象$_7$" in Fig. 1, then SEP_k will be "No"; if the current $\overline{s}_{a(k)}$ is "$\circ_8$'' then SEP_k will be "Yes". It is introduced

because SCM and TCM for a sentence-end-punctuation construction are very likely to be "Same" regardless of other features. Therefore, to avoid introducing bias, it is used to distinguish this special case from the other cases.

- *TM candidate set status* (CSS): The CSS restricts the possible status of $tm\_\bar{t}_{a(k)}$ and is a member of {Single, Left-Ext, Right-Ext, Both-Ext, NA}. Where "Single" means that there is only one candidate $tm\_\bar{t}_{a(k)}$ for the given source phrase $tm\_\bar{s}_{a(k)}$; "Left-Ext" means that there are multiple $tm\_\bar{t}_{a(k)}$ candidates, where all the candidates are generated by extending only the left boundary; "Right-Ext" means that all the candidates are generated by only extending to the right; "Both-Ext" means that both sides have been extended; and "NA" means that $tm\_\bar{t}_{a(k)}$ is null. For the TM source phrase $tm\_\bar{s}_{a(k)}$ "关联$_4$ 的$_5$ 对象$_6$" in Fig. 1, the linked TM target phrase $tm\_\bar{t}_{a(k)}$ is "object$_2$ that$_3$ is$_4$ associated$_5$", and there are five other candidates as a result of extending to both sides. Therefore, its associated CSS_k is "Both-Ext".
- *TM candidate relative length status* (CRL): The CRL indicates the relative length status of the given $tm\_\bar{t}_{a(k)}$ and is a member of {Original, Left-Longest, Right-Longest, Both-Longest, Medium, NA}. Where "Original" means that the given $tm\_\bar{t}_{a(k)}$ is the original one without extension; "Left-Longest" means that the given $tm\_\bar{t}_{a(k)}$ is extended only from the left and is the longest one; "Right-Longest" means that the given $tm\_\bar{t}_{a(k)}$ is extended only from the right and is the longest one; "Both-Longest" means that the given $tm\_\bar{t}_{a(k)}$ is extended from both sides and is the longest one; "Medium" means that the given $tm\_\bar{t}_{a(k)}$ has been extended but is not the longest one; "NA" means that the current $tm\_\bar{t}_{a(k)}$ is null.
- *Target phrase adjacent candidate relative position matching status* (RPM): The RPM indicates the matching status between the relative position of $[\bar{t}_{k-1}, \bar{t}_k]$ and the relative position of $[tm\_\bar{t}_{a(k-1)}, tm\_\bar{t}_{a(k)}]$. It checks if $[\bar{t}_{k-1}, \bar{t}_k]$ is positioned in the same order as $[tm\_\bar{t}_{a(k-1)}, tm\_\bar{t}_{a(k)}]$ and reflects the quality of the ordering of the given target candidate $\bar{t}_k$. RPM is a member of {Adjacent-Same, Adjacent-Substitute, Linked-Interleaved, Linked-Cross, Linked-Reversed, Skip-Forward, Skip-Cross, Skip-Reversed, NA}. The detailed specification of RPM is given at Appendix B.

Eq. (5) is derived with the assumption that RPM_k is independent of CRL_k, CSS_k, SPL_k and SEP_k, because CRL and CSS are mainly used to address word insertions (which are not closely related to reordering). In addition, SPL and SEP will also be ignored during the evaluation of the probability of RPM_k because the length of the source phrase would not substantially affect the reordering, and SEP is used to distinguish the sentence punctuation from other phrases.

4.1.2. Experiments and results

To fairly compare the SMT with the TM, we use the same training corpora for both the SMT and TM systems. Furthermore, the sentences in the development set and the test set are divided into various intervals according to their associated best fuzzy match scores, which are obtained by matching each sentence in the development/test with various sentences in the training-set and then taking the maximum value. Corpus statistics for the TM database (also the SMT training set) are shown in Table 2, and the statistics for each interval in the development set and the test set are shown in Table 3.

In comparison to the TM system in the interval [0.9, 1.0), it can be observed that the SMT is slightly better than the TM based on the BLEU score (81.4 vs. 81.3). However, the TM significantly exceeds the SMT based on the TER score (9.8 vs. 13.0). These results illustrate why professional translators prefer to adopt TM rather than SMT for post-editing. Additionally, when the fuzzy match score decreases, the translation performances of both the TM and the SMT systems decrease. Moreover, the performance of the TM deteriorates much faster than the SMT does, because there are more unmatched parts when the fuzzy match score is low.

Tables 4 and 5 also show that Model-I achieves not only the best BLEU score (56.5) but also the best TER score (33.3) across all intervals (shown by the last row in the table). To be more specific, compared with the TM and the SMT systems, Model-I significantly outperforms both the TM and the SMT systems in both BLEU and TER when

Table 2
Corpus statistics of the TM database and the SMT training set when they are the same.

	#Sentences	#Chn. words	#Chn. VOC.	#Eng. words	#Eng. VOC.
TM dataset/SMT training set	261,906	3,623,516	43,112	3,627,028	44,221

Table 3
FMS interval statistics of the development set and the test set when the TM database and the SMT training set are the same.

Intervals	Development set			Test set		
	#Sentences	#Words	W/S	#Sentences	#Words	W/S
[0.9, 1.0)	295 (11%)	4816	16.3	269 (10%)	4468	16.6
[0.8, 0.9)	387 (15%)	5144	13.3	362 (14%)	5004	13.8
[0.7, 0.8)	296 (12%)	3940	13.3	290 (11%)	4046	14.0
[0.6, 0.7)	349 (14%)	4555	13.1	379 (15%)	4998	13.2
[0.5, 0.6)	465 (18%)	6151	13.2	472 (18%)	6073	12.9
[0.4, 0.5)	378 (15%)	5628	14.9	401 (16%)	5921	14.8
[0.3, 0.4)	309 (12%)	5892	19.1	305 (12%)	5499	18.0
(0.0, 0.3)	90 (4%)	2459	27.3	98 (4%)	2639	26.9
(0.0, 1.0)	2569 (100%)	38,585	15.0	2576 (100%)	38,648	15.0

Table 4
Translation results[a] (BLEU) of various approaches when the TM database and the SMT training set are the same.

Intervals	TM	SMT	Model-I	Koehn-10	Ma-11-U
[0.9, 1.0)	81.3	81.4	89.4*#	82.8	82.8
[0.8, 0.9)	73.7	76.2	84.0*#	79.2 *	77.7
[0.7, 0.8)	63.6	67.7	74.7*#	71.0 *	69.8
[0.6, 0.7)	43.6	54.6	57.5*#	53.1	56.4
[0.5, 0.6)	27.4	46.3	47.5*#	39.3	47.7
[0.4, 0.5)	15.4	37.2	38.2*#	29.0	37.9
[0.3, 0.4)	8.2	29.3	29.2#	23.6	30.2
[0.0, 0.3)	4.1	26.4	25.6#	18.6	26.9
(0.0, 1.0)	40.1	53.0	56.5*#	50.3	54.3

[a] "*" indicates that it is significantly better than both the TM and the SMT baselines, and "#" indicates that it is significantly better than Koehn-10.

Table 5
Translation results[a] (TER) of various approaches when the TM database and the SMT training set are the same (see footnote 9).

Intervals	TM	SMT	Model-I	Koehn-10	Ma-11-U
[0.9, 1.0)	9.8	13.0	**6.8*#**	13.0	11.9
[0.8, 0.9)	16.2	16.	**10.8*#**	15.3 *	14.7
[0.7, 0.8)	27.8	22.8	**17.1*#**	21.9 *	21.1
[0.6, 0.7)	46.4	33.4	**30.0*#**	35.9	31.8
[0.5, 0.6)	62.6	39.6	38.7*#	47.4	38.0
[0.4, 0.5)	73.9	47.2	**46.0*#**	56.8	46.1
[0.3, 0.4)	79.9	55.7	55.9#	64.6	54.2
[0.0, 0.3)	85.3	**61.8**	63.5#	73.3	60.7
(0.0, 1.0)	50.5	35.9	**33.3*#**	40.8	34.5

[a] "*" indicates that it is significantly better than both the TM and the SMT baselines, and "#" indicates that it is significantly better than Koehn-10.

the fuzzy match score is above 0.4. All these improvements show that our integrated models have combined the strength of both TM and SMT.

However, the improvements from Model-I decrease when the fuzzy match score decreases. For example, Model-I outperforms SMT by 8.0 BLEU points in the interval [0.9, 1.0), while the advantage is only 2.9 BLEU points in the

Source	如果 0 禁用 1 此 2 策略 3 设置 4 ， 5 internet6 explorer7 不 8 搜索 9 internet10 查找 11 浏览器 12 的 13 新 14 版本 15 ， 16 因此 17 不 18 会 19 提示 20 用户 21 安装 22 。 23
Reference	if0 you1 disable2 this3 policy4 setting5 ,6 internet7 explorer8 does9 not10 check11 the12 internet13 for14 new15 versions16 of17 the18 browser19 ,20 so21 does22 not23 prompt24 users25 to26 install27 them28 .29
TM Source	如果 0 不 1 配置 2 此 3 策略 4 设置 5 ， 6 internet7 explorer8 不 9 搜索 10 internet11 查找 12 浏览器 13 的 14 新 15 版本 16 ， 17 因此 18 不 19 会 20 提示 21 用户 22 安装 23 。 24
TM Target	if0 you1 do2 not3 configure4 this5 policy6 setting7 ,8 internet9 explorer10 does11 not12 check13 the14 internet15 for16 new17 versions18 of19 the20 browser21 ,22 so23 does24 not25 prompt26 users27 to28 install29 them30 .31 **[Two deletions, one substitution]**
TM Alignment	0-0 1-3 2-4 3-5 4-6 5-7 6-8 7-9 8-10 9-11 11-15 13-21 14-19 15-17 16-18 17-22 18-23 19-24 21-26 22-27 23-29 24-31
SMT	if you disable this policy setting , internet explorer does not ***prompt users to install*** internet for new versions of the browser . **[Miss 7 target words: 9~12, 20~21, 28; and has one wrong permutation]**
Koehn-10	if you do you disable this policy setting , internet explorer does not check the internet for new versions of the browser , so does not prompt users to install them . **[Insert two spurious target words]**
Ma-11-U	if you disable this policy setting , internet explorer does not ***prompt users to install*** internet for new versions of the browser . **[Miss 7 target words: 9~12, 20~21, 28; Has one wrong permutation]**
Model-III	if you disable this policy setting , internet explorer does not check the internet for new versions of the browser , so does not prompt users to install them . **[Exactly the same as the reference]**

Fig. 4. A translation example in the interval [0.9, 1.0) (with FMS = 0.920) when the TM database and the SMT training set are the same.

interval [0.6, 0.7). This is because a lower fuzzy match score implies that there are more unmatched parts between *s* and *tm_s*; the output of TM is thus less reliable.

Compared with the Koehn-10 and the Ma-11-U systems, it is observed that Model-I performs significantly better than Koehn-10 in all intervals. More importantly, the proposed model achieves much better TER scores than the TM system does even at in the interval [0.9, 1.0); in contrast, Koehn-10 cannot exceed the TM system in this interval. Moreover, Model-I performs much better than Ma-11-U does in most intervals. Therefore, it can be concluded that the proposed Model-I significantly outperform the pipeline approaches in this task.[9]

Fig. 4 gives an example in the interval [0.9, 1.0) that shows the difference among different system outputs. It can be observed that "you do" is redundant for Koehn-10 because they are insertions and thus are kept in the XML input. However, the SMT system still inserts another "you" despite "you do" already existing. This problem does not occur for Ma-11-U, but it misses some words and adopts one wrong permutation. However, Model-I produces a perfect translation because it considers both TM content and TM position information.

4.2. Adopting a different SMT training set from the same domain

In the last section, we evaluate the performance of various approaches when the TM database and the SMT training set are the same. However, the TM database might be empty when we start the translation task, and it is growing up while translators keep adding more sentence pairs to the TM database. As a SMT system is

[9] Wang et al. (2013) have shown that this approach also behaves similarly when the original corpus is replaced by Hong Kong Hansards corpus (LDC2004T08).

usually trained on a large data set and the required training-time often takes several days, online learning is not supported by most SMT systems. Therefore, we cannot always include the whole TM database to train the SMT system from time to time while the TM database keeps changing. Therefore, the following Model-II is proposed to handle this scenario, in which the TM database and the SMT training set are different but from the same domain.

4.2.1. Proposed Model-II

As mentioned in Section 2.3, if the TM database and the SMT training set are the same, it is not necessary to add those TM matched phrase pairs into the SMT phrase table. On the contrary, when the TM database and the SMT training set are different, we need to merge those matched TM phrase pairs into the SMT phrase table for each input sentence. However, the Model-I does not distinguish the newly added TM phrase pairs from those original SMT phrase pairs in $P(M_k|L_k,\ z)$. Therefore, we add two indicators, *Target Phrase Origin* (TPO) (mainly for the category (2) in Section 2.3) and *Source Phrase Origin* (SPO) (mainly for the category (3) in Section 2.3), which are members of {Original, Newly-Added}, to favor the TM phrases as follows:

$$\begin{aligned} & P(M_k|L_k,z) \\ & \triangleq P([RPM,TCM,CRL]_k|[SCM,NLN,CSS,SPL,SEP]_k,z) \\ & \approx \left\{ \begin{array}{r} P(TCM_k|SCM_k,NLN_k,CRL_k,SPL_k,SEP_k,z) \\ \times P(CRL_k|CSS_k,SCM_k,NLN_k,SEP_k,z) \\ \times P(RPM_k|TCM_k,SCM_k,NLN_k,z) \\ \times P(TPO_k|SPO_k,NLN_k,z) \end{array} \right\} \end{aligned} \tag{6}$$

In the above derivation, we assume that TPO_k is independent of RPM_k, TCM_k, CRL_k, SCM_k, CSS_k, SPL_k and SEP_k. This would be our proposed Model-II. Also, different from the Model-I, in order to distinguish these new phrase pairs from the original SMT phrase pairs, eight additional feature weights λ_m for the translation probabilities (lexicon and phrase transfer in both directions) and two more feature weights for the phrase penalty will be adopted.

4.2.2. Experiments and results

Under this scenario, the training set mentioned in Section 3.1 is randomly divided into two equal parts. The first part is adopted as the TM database, while the second part is adopted as the SMT training set. The development and the test sets are still the same as those of the last scenario. The detailed corpus statistics are shown in Table 6. Since, the TM database has changed, the statistics in each interval are different from those in the last section. The FMS interval statistics for the development set and the test set are thus re-shown in Table 7.

Compared with the experiment from the first scenario, there are fewer sentences in the high fuzzy match intervals and more sentences in the low fuzzy match intervals in this experiment, as shown in Table 7. This is because the TM database has been reduced to half of its original size. Therefore, the distribution of FMS intervals leans to the side with low fuzzy match scores.

Since there are insufficient samples to estimate the translation probabilities for those newly added TM phrases, we use the following method to assign the translation probabilities. For those TM phrase pairs of the second category mentioned in Section 2.3, because their source phrases have already existed in the SMT phrase table, there will be 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 the TM phrase pairs of the third category, because there are no corresponding phrase pairs existing in the SMT phrase table, we will simply

Table 6
Corpus statistics for the TM database and the SMT training set when they are different but from the same domain.

	#Sentences	#Chn. words	#Chn. VOC.	#Eng. words	#Eng. VOC.
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 7
FMS interval statistics of the development and the test sets when the TM database and the SMT training set are different but from the same domain.

Intervals	Develop			Test		
	#Sentences	#Words	W/S	#Sentences	#Words	W/S
[0.9, 1.0)	182 (7%)	2951	16.2	147 (6%)	2431	16.5
[0.8, 0.9)	249 (10%)	3239	13.0	255 (10%)	3438	13.5
[0.7, 0.8)	252 (10%)	3247	12.9	244 (9%)	3299	13.5
[0.6, 0.7)	302 (12%)	3831	12.7	355 (14%)	4674	13.2
[0.5, 0.6)	535 (21%)	7044	13.2	488 (19%)	6125	12.6
[0.4, 0.5)	478 (19%)	6851	14.3	514 (20%)	7525	14.6
[0.3, 0.4)	417 (16%)	7380	17.7	419 (16%)	7082	16.9
(0.0, 0.3)	154 (6%)	4402	26.2	154 (6%)	4072	26.5
(0.0, 1.0)	2,569 (100%)	38,585	15.0	2576 (100%)	38,648	15.0

Table 8
Translation results[a] (BLEU) of various approaches according to whether merging TM phrase pairs or not when the TM database and the SMT training set are different but from the same domain.

Intervals	SMT	SMT+	Model-I	Model-I+
[0.9, 1.0)	63.7	73.6+	80.7	**86.4+**
[0.8, 0.9)	60.8	74.0+	79.0	**83.4+**
[0.7, 0.8)	60.6	65.5+	68.6	**71.4+**
[0.6, 0.7)	53.4	56.1+	55.6	**57.8+**
[0.5, 0.6)	45.6	47.0+	47.4	**48.4+**
[0.4, 0.5)	41.8	42.0	**42.6**	42.3
[0.3, 0.4)	36.0	35.5	**36.1**	35.3
(0.0, 0.3)	32.6	33.2	33.5	**33.2**
(0.0, 1.0)	46.7	49.4+	51.0	**52.3+**

[a] "+" indicates that those newly added TM phrases significantly ($p < 0.05$) improve the translation results ("SMT" vs. "SMT+" and "Model-I" vs. "Model-I+").

assign the probability "1.0"[10] to them for their associated four SMT translation probabilities. Furthermore, to distinguish these newly added phrase pairs (i.e., in the second and third categories) from the original SMT phrase pairs, we use eight additional feature weights λ_m for the translation probability and two additional 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 the SMT and the Model-I. Table 8 shows the translation results using the BLEU score. "SMT" and "Model-I" denote that we do not merge the TM phrase pairs into the SMT phrase table during the decoding. In other words, they only use the original SMT phrase table. "SMT+" and "Model-I+" mean that we dynamically merge the TM phrase pairs into the SMT phrase table. In this table, "+" indicates that those newly added TM phrases significantly ($p < 0.05$) improve the translation results ("SMT"vs."SMT+" and "Model-I"vs."Model-I+").

It can be observed that adding TM phrase pairs significantly improves the translation results when the fuzzy match score is above 0.5. For example, in the interval [0.9, 1.0), those added TM phrase pairs significantly improve the SMT system from 63.7 to 73.6 and improve the score of the Model-I from 80.7 to 86.4. However, comparing the Model-I to the Model-I+, the improvement from merging the TM phrase pairs decreases when the fuzzy match score decreases because the ratio of matched TM sub-segments becomes smaller in low fuzzy match intervals.

Additionally, using the same original SMT phrase table, the Model-I performs better than the SMT system in each interval. For example, in the interval [0.9, 1.0), the TM information significantly improves the translation result from 63.7 to 80.7. This shows that the TM information is very useful. On the other hand, although the SMT+ greatly outperforms the SMT in each interval, the Model-I+ still significantly outperforms the SMT+ in most intervals.

[10] This value is not important as its associated weight will be tuned later.

Table 9
Translation results[a] (BLEU) of various approaches when the TM database and the SMT training set are different but from the same domain.

Intervals	TM	SMT+	Model-I+	Model-II	Koehn-10	Ma-11-U
[0.9, 1.0)	79.9	73.6	86.4*#	**86.7***#	82.2	67.6
[0.8, 0.9)	72.7	74.0	83.4*#	**83.4***#	79.5*	67.0
[0.7, 0.8)	59.6	65.5	71.4*#	**72.1***#	67.5	62.6
[0.6, 0.7)	41.6	56.1	57.8*#	**58.7***#&	51.8	56.7
[0.5, 0.6)	25.2	47.0	**48.4** *#	48.3*#	39.1	47.9
[0.4, 0.5)	14.6	42.0	42.3#	**43.0***#&	31.6	42.9
[0.3, 0.4)	7.5	**35.5**	35.3#	35.3#	25.3	36.3
(0.0, 0.3)	4.9	33.2	**33.2#**	**33.2#**	23.7	33.1
(0.0, 1.0)	31.1	49.4	52.3*#	**52.6***#&	44.3	48.9

[a] "*" indicates that it is significantly better than both the TM and the SMT baselines; "#" indicates that it is significantly better than Koehn-10; and "&" indicates that they are significantly better ($p < 0.05$) than Model-I+.

Therefore, the benefit of utilizing the TM information in the model and the benefit of adding TM phrase pairs can be *jointly enjoyed*. Taking the interval [0.9, 1.0) as an example, the added TM phrase pairs first improve the translation results from 63.7 (SMT) to 73.6 (SMT+), and subsequently, the TM information further increases it to 86.4 (Model-I+).

To show the effectiveness of the Model-II, Table 9 presents the translation results of the TM, the SMT+, the other two baselines (Koehn-10 and Ma-11-U), and two proposed models (the Model-I+ and the Model-II). Scores marked with "*" indicate that they are significantly better ($p < 0.05$) than both the TM and the SMT+ baselines; scores marked with "&" indicates that they are significantly better ($p < 0.05$) than the Model-I+; and the other formats are the same as those adopted in Tables 4 and 5. Additionally, please note that the performance of the TM cannot be directly compared with that of the SMT+ in each interval because FMS is measured against the TM database but not against the SMT training set (i.e., the sentence in the interval [0.9, 1.0) might not also possess high FMS if it is measured against the SMT training set).

Compared with the TM and the SMT+ systems, the Model-I+ is significantly better for BLEU (also for TER, which is not shown here. Since all the obtained TER results are highly correlated with that of BLEU, they will be skipped to save space from now on) when the fuzzy match score is above 0.5. In addition, the Model-II outperforms both the TM and the SMT+ systems for BLEU when the fuzzy match score is above 0.4. Furthermore, the improvements from the proposed models decrease when the fuzzy match score decreases because the TM information is less reliable in low fuzzy match intervals. All the trends observed are similar to those in the last sub-section.

Across all intervals (the last row in the table), the Model-II achieves the best BLEU score (52.6). In intervals where the fuzzy match score is above 0.4, the Model-I+ and the Model-II are the best models for the BLEU score. In addition, the Model-II performs slightly better than the Model-I+ in most intervals. However, both the Model-I+ and the Model-II achieve significant improvements over the TM and the SMT+ systems. Furthermore, it can be observed that both the Model-I+ and the Model-II outperform Koehn-10 for BLEU score in each interval. Additionally, both the Model-I+ and the Model-II are much better than Ma-11-U in most intervals. Therefore, it can be concluded that the proposed models significantly outperform the pipeline approaches in this scenario.

To better understand our model behavior, Figs. 5 and 6 give two translation examples in the interval [0.9, 1.0) and demonstrate the effectiveness of the newly added TM phrase pairs. Example 1 in Fig. 5 shows that those phrases alleviate the translation problem of unseen phrases. In example 1, "表述$_2$" is an out-of-vocabulary (OOV) word. Therefore, the SMT system and Model-I cannot translate it. However, it is a matched word between the input and the TM source sentence. Therefore, we can extract many unseen phrases that can cover "表述$_2$", such as (1) 【表述$_2$ → rephrase$_0$】; (2) 【重新$_1$ 表述$_2$ → rephrase$_0$】; (3) 【请$_0$ 重新$_1$ 表述$_2$ 您$_3$ 的$_4$ 搜索$_5$ → rephrase$_0$ your$_1$ search$_2$】; (4)【请$_0$ 重新$_1$ 表述$_2$ 您$_3$ 的$_4$ 搜索$_5$ 查询$_6$ → rephrase$_0$ your$_1$ search$_2$ query$_3$】; (5)【请$_0$ 重新$_1$ 表述$_2$ 您$_3$ 的$_4$ 搜索$_5$ 查询$_6$ → rephrase$_0$ your$_1$ search$_2$ query$_3$ to$_4$】 and so on. All these phrase pairs would be dynamically added to the SMT phrase table.

It is observed that the SMT+ system, the Model-I+ and the Model-II have all correctly translated the associated OOV word in this case. However, they adopt different TM phrase pairs. Both the SMT+ and the Model-II select 【请$_0$ 重新$_1$ 表述$_2$ 您$_3$ 的$_4$ 搜索$_5$ 查询$_6$ → rephrase$_0$ your$_1$ search$_2$ query$_3$】, whereas the Model-I+ chooses

Example 1 (FMS = 0.962)	
Source	请 0 重新 1 表述 2 您 3 的 4 搜索 5 查询 6 以 7 仅 8 使用 9 一 10 种 11 运算符 12 :13 “14 且 15 ”16 、 17 “18 或 19 ”20 或 21 引号 22 (23 “24)25 。 26
Reference	rephrase$_0$ your$_1$ search$_2$ query$_3$ to$_4$ use$_5$ only$_6$ one$_7$ kind$_8$ of$_9$ operator$_{10}$:$_{11}$ and$_{12}$,$_{13}$ or$_{14}$,$_{15}$ or$_{16}$ quotation$_{17}$ marks$_{18}$ (19 “20)21 .22
TM Source	请 0 重新 1 表述 2 您 3 的 4 搜索 5 查询 6 并 7 仅 8 使用 9 一 10 种 11 运算符 12 :13 “14 且 15 ”16 、 17 “18 或 19 ”20 或 21 引号 22 (23 “24)25 。 26
TM Target	rephrase$_0$ your$_1$ search$_2$ query$_3$ to$_4$ use$_5$ only$_6$ one$_7$ kind$_8$ of$_9$ operator$_{10}$:$_{11}$ and$_{12}$,$_{13}$ or$_{14}$,$_{15}$ or$_{16}$ quotation$_{17}$ marks$_{18}$ (19 “20)21 .22 **[Exactly the same as the reference]**
TM Alignment	2-0 3-1 5-2 6-3 8-6 9-5 10-7 11-8 12-10 13-11 15-12 17-13 19-14 20-15 21-16 22-17 23-19 24-20 25-21 26-22
SMT	please restart |0-1| : “ |13-14| and |15-15| “ , “ |16-18| or |19-19| “ or |20-21| quotes (“) |22-25| 表述 |2-2| your search query |3-6| to only |7-8| one |9-11| operator |12-12| . |26-26| **[One OOV word; One error word: 0; multiple wrong permutations]**
SMT$^+$	rephrase your search query |0-6| to |7-7| use only one kind of |8-11| operator : |12-13| and |14-15| , or , |16-20| or quotation |21-22| marks (“) . |23-26| **[Exactly the same as the reference]**
Model-I	please reinstall |0-1| to |7-7| ‘ |14-14| and |15-15| , |16-18| or |19-19| , or |20-21| quotation marks |22-22| (|23-23| “) . |24-26| your search query |3-6| to use |9-9| only |8-8| one |10-10| kind of |11-11| operator |12-12| : |13-13| 表述 |2-2| **[One OOV word; One error word: 0; multiple wrong permutations]**
Model-I+	rephrase your search |0-5| query to |6-7| use only one kind of |8-11| operator : and , |12-17| or , or |18-21| quotation marks (“) . |22-26| **[Exactly the same as the reference]**
Model-II	rephrase your search query |0-6| to |7-7| use only one kind of operator : |8-13| and , |14-17| or , or |18-21| quotation marks (") . |22-26| **[Exactly the same as the reference]**

Fig. 5. Translation example in the interval [0.9, 1.0) when the TM database and the SMT training set are different but from the same domain.

【请$_0$ 重新$_1$ 表述$_2$ 您$_3$ 的$_4$ 搜索$_5$ → rephrase$_0$ your$_1$ search$_2$】. Both are not the longest phrase pairs. Therefore, it is difficult to choose the best TM phrase pair before decoding; based on this reason, we thus provide multiple TM phrases as candidates.

Those unseen phrases in the example 1 are results of OOV word. However, there would be unseen phrases even when no OOV word exists. In the example 2 of Fig. 6, all the source words in 【这个$_0$ 区域$_1$ 中$_2$ 的$_3$ 设置$_4$】, 【站点$_{10}$ 区域$_{11}$ 处于$_{12}$ “$_{13}$ 锁定$_{14}$】 and 【锁定$_{14}$ ”$_{15}$ 模式$_{16}$ 时$_{17}$ ($_{18}$ 例如$_{19}$】 are not OOV words. However, this source phrase is unseen in the original SMT phrase table. With the help of TM phrase pairs 【这个$_0$ 区域$_1$ 中$_2$ 的$_3$ 设置$_4$ → the$_0$ settings$_1$ in$_2$ this$_3$ zone$_4$】, 【站点$_{10}$ 区域$_{11}$ 处于$_{12}$ “$_{13}$ 锁定$_{14}$ → zone$_{10}$ is$_{11}$ in$_{12}$ ’$_{13}$ lockdown$_{14}$】 and 【锁定$_{14}$ ”$_{15}$ 模式$_{16}$ 时$_{17}$ ($_{18}$ 例如$_{19}$ → lockdown$_{14}$ ’$_{15}$ mode$_{16}$,$_{17}$ such$_{18}$】, both the Model-I+ and the Model-II achieve better translation results.

Example 2 (FMS = 0.968)	
Source	这个 0 区域 1 中 2 的 3 设置 4 只有 5 在 6 受 7 限制 8 的 9 站点 10 区域 11 处于 12 “13 锁定 14 ”15 模式 16 时 17 (18 例如 19 网络 20 协议 21 锁定 22 安全 23 功能 24 生效 25 时 26)27 才 28 适用 29 。 30
Reference	the0 settings1 in2 this3 zone4 apply5 only6 if7 the8 restricted9 zone10 is11 in12 '13 lockdown14 '15 mode16 ,17 such18 as19 when20 the21 network22 protocol23 lockdown24 security25 feature26 is27 in28 effect29 .30
TM Source	这个 0 区域 1 中 2 的 3 设置 4 只有 5 在 6 受 7 信任 8 的 9 站点 10 区域 11 处于 12 “13 锁定 14 ”15 模式 16 时 17 (18 例如 19 网络 20 协议 21 锁定 22 安全 23 功能 24 生效 25 时 26)27 才 28 适用 29 。 30
TM Target	the0 settings1 in2 this3 zone4 apply5 only6 if7 the8 trusted9 zone10 is11 in12 '13 lockdown14 '15 mode16 ,17 such18 as19 when20 the21 network22 protocol23 lockdown24 security25 feature26 is27 in28 effect29 .30 **[One wrong word: 9]**
TM Alignment	0-3 1-4 2-2 3-0 4-1 5-6 8-9 11-10 12-11 13-13 14-14 15-15 16-16 19-18 20-22 21-23 22-24 23-25 24-26 25-29 26-20 30-30
SMT	applies |28-29| only |5-6| when |17-17| the network protocol lockdown security |20-23| feature |24-24| is in |12-12| effect |25-25| when a |26-27| restricted sites zone |7-11| settings in |2-4| this |0-0| zone |1-1| (for example , |18-19| " locked |13-15| mode |16-16| . |30-30| **[Six wrong words: 5, 13~15, 18, 19; missing two words: 0, 7; multiple wrong permutations]**
SMT+	only if the |5-7| zone is |9-12| locked |13-15| mode , such as |16-19| when the |26-29| network protocol lockdown security |20-23| feature is in effect |24-25| limit |8-8| the settings in this zone |0-4| . |30-30| **[Four wrong words: 5, 14, 18, 19; missing three word: 5, 13, 15; multiple wrong permutations]**
Model-I	the settings in |2-4| this |0-0| zone |1-1| is in |12-12| effect |25-25| when the |26-26| network protocol lockdown security feature is |20-24| only |5-6| applicable |28-29| when |17-17| the restricted |7-8| sites |9-10| zone |11-11|) |27-27| ' |13-13| lockdown |14-14| ' mode |15-16| , such as |18-19| . |30-30| **[One wrong word: 9; insert one spurious target word; multiple wrong permutations]**
Model-I+	the settings in this zone |0-4| apply only if the |5-7| limit |8-8| zone is in ' |9-13| lockdown ' mode , such |14-19| as when |26-29| the network protocol lockdown security |20-23| feature is in effect |24-25| . |30-30| **[One wrong word: 9]**
Model-II	the settings in this zone |0-4| apply only if the |5-7| restricted |8-9| zone is in ' lockdown |10-14| ' mode , such |15-19| as when |26-29| the network protocol lockdown security |20-23| feature is in effect |24-25| . |30-30| **[Exactly the same as the reference]**

Fig. 6. Another translation example in the interval [0.9, 1.0) when the TM database and the SMT training set are different but from the same domain.

To further verify the proposed models in this case, we swap the TM database and the SMT training set and re-run the experiments. Similar improvements are still observed: both the Model-I+ and the Model-II achieve significant improvements over the TM and the SMT+ systems. All these results have shown that the proposed models are robust.

In a real environment, the SMT training corpora and the TM database are the same before translation starts. However, the TM database will gradually deviate from the SMT training set as the translation task progresses. In fact, Sections 4.1 and 4.2 represent two extreme cases for real applications (i.e., the case that is at the beginning, and the case that is approaching the end of the translation task). Because our proposed models significantly outperform the baselines and the previous approaches in both cases, our proposed models are shown to be superior in real applications.

4.3. Adopting a different cross-domain SMT training set

In the previous two sub-sections, we evaluate the performances of various approaches when the TM database and the SMT training set are from the same domain. However, the TM database is usually not large enough to train an SMT system when it is applied to a technical domain other than the news domain. In addition, some professional translators even are not willing to expose the whole TM database to the SMT system providers (Cancedda, 2012). In this situation, we will be forced to first offline train a SMT model on an *out-domain* (usually the news domain) that possesses a substantial amount of training data and then fix the obtained phrase-based SMT model. Afterwards, we incorporate it online with an additional TM database that is from another *in-domain*.

Basically, the adopted approach will work when the SMT can provide good enough output in the intervals with high fuzzy matching ratio (please see the footnote 13 at page 21). Since the fuzzy matching ratio is measured against TM database, not against SMT training data, our approach is not sensitive to which corpus is adopted as the SMT training-set once the above condition is met. Adopting another TM database only changes the percentages of those high fuzzy matching ratio intervals in the test set. This statement is supported by Wang et al. (2013), which has shown that the behavior of our approach is the same when the corpus of CWMT2009, instead of original Hong Kong hansards corpus (LDC2004T08), is adopted.

4.3.1. Proposed Model-III (with only the Top-1 TM sentence-pair)

The above mentioned Model-I and Model-II do not explicitly favor the situation that a target phrase candidate $\bar{t}_k$ *exactly* matches a $tm\_\bar{t}_{a(k)}$ candidate; however, according to our observations, a $\bar{t}_k$ of this type is usually preferred in this scenario. Therefore, we add a feature called *Same TM Target Phrase Existence Indicator* (STMI), which is a member of {Yes, No}, Where "Yes" means that the given $\bar{t}_k$ exactly matches one available $tm\_\bar{t}_{a(k)}$ candidate and "No" indicates that the given $\bar{t}_k$ does not exactly match any $tm\_\bar{t}_{a(k)}$ candidate in the TM target candidates set. With this new feature, $P(M_k|L_k, z)$ can be derived as follows.

$$
\begin{aligned}
&P(M_k|L_k,z)\\
&\triangleq P([TPO,RPM,TCM,CRL,STMI]_k|[SCM,SPO,NLN,CSS,SPL,SEP]_k,z)\\
&\approx \left\{ \begin{array}{r} P(TCM_k|SCM_k,NLN_k,CRL_k,SPL_k,SEP_k,STMI_k,z)\\ \times P(CRL_k|CSS_k,SCM_k,NLN_k,SEP_k,z)\\ \times P(RPM_k|TCM_k,SCM_k,NLN_k,z)\\ \times P(TPO_k|SPO_k,NLN_k,z)\\ \times P(STMI_k|SCM_k,NLN_k,z) \end{array} \right\}
\end{aligned}
\tag{7}
$$

In the above derivation, we assume that TPO_k, CRL_k, and RPM_k are independent of $STMI_k$ and that $STMI_k$ is independent of SPO_k, CSS_k, SPL_k and SEP_k. This would be the proposed Model-III. To distinguish those new phrase pairs from the original SMT phrase pairs, we adopt the same strategy specified in the Model-II (i.e., adding eight additional feature weights λ_m for the translation probabilities and two feature weights for the phrase penalties).

4.3.2. Experiments and results of Model-III

To evaluate the cross-domain performance, we adopt a news corpora about computers and science[11] from CWMT09 (Liu and Zhao, 2009) as the SMT training set and subsequently adopt the computer technical database

[11] We have also tested the news corpus from other domains such as political news. Because the performance of phrase-based SMT drops dramatically in those cases (mainly due to many unseen phrase pairs and n-gram mismatches), combining the TM and the SMT thus brings no benefit. As the result, the proposed models only achieve a small improvement or are worse than the TM system in the high fuzzy match intervals even with the aid of the SMT.

Table 10
Corpus statistics for the TM database and the SMT training set when they are from different domains.

	#Sentences	#Chn. words	#Chn. VOC.	#Eng. words	#Eng. VOC.
TM dataset	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

(specified in Section 3.1) as the TM database. The SMT training set contains approximately 404k bilingual sentence pairs, which includes approximately 9M Chinese words and 8.7M English words. The corpus statistics are shown in Table 10. Because the TM database and the test set (also the development set) are the same as those in Section 4.1.2, the statistics in each interval are the same as those in Table 3.

The adopted training procedure is the same as that mentioned in the previous experiment. Table 11 presents the translation results of the TM, the SMT, the SMT+, and the four proposed models (Model-I, Model-I+, Model-II and Model-III). In this table, "+" indicates that those newly added TM phrases significantly ($p < 0.05$) improve the translation results ("SMT+" vs. "SMT" and "Model-I+" vs. "Model-I"), scores marked with "*" indicate that they are significantly better ($p < 0.05$) than both the TM and the SMT+ baselines, "&" means that they are significantly better ($p < 0.05$) than the Model-I+, and "$" means that they are significantly better ($p < 0.05$) than the Model-II. The bold entries are the best result in that interval.

Comparing the TM system with the SMT system, the performance of the in-domain TM system significantly exceeds that of the out-domain SMT system. Because the fuzzy match intervals are divided according to the TM database, not according to the SMT training-set, the translation result of the SMT system in the interval [0.8, 0.9) slightly outperforms that in the interval [0.9, 1.0). In addition, adding TM phrase pairs significantly improves the translation results when the fuzzy match score is above 0.5 (comparing the SMT+ with the SMT, and the Model-I+ with the Model-I). Furthermore, all those trends observed (such as that the benefits of utilizing TM information and adding TM phrase pairs can be jointly enjoyed) are similar to that of the last scenario.

In comparison with the TM and the SMT+ systems, the Model-I+ (also the Model-II and the Model-III) achieves better translation results in most intervals for the BLEU score when the fuzzy match score is above 0.5. In addition, the Model-II is slightly better than the Model-I+, which is similar to what was observed in the last scenario. Furthermore, the Model-III not only achieves the best overall BLEU score (48.8) but also significantly outperforms the TM system, the SMT+ system, the Model-I+ and the Model-II when the fuzzy match score is above 0.5.

In Model-III, except for the TPO factor, all other factors are trained on the SMT training set (out-domain). Because the development and the test sets are from the in-domain data, there is a domain-mismatch problem for the TM factors. Generally, in the technical domain, which is suitable for TM applications, the translations (especially for technical terms) are more consistent than those in the news domain. In other words, the same source phrase in various places tends to have exactly the same translation in technical domains. Therefore, we use a log-linear

Table 11
Translation results[a] (BLEU) of various approaches when the TM database and the SMT training set are from different domains.

Intervals	TM	SMT	SMT+	Model-I	Model-I+	Model-II	Model-III
[0.9, 1.0)	81.3	30.9	64.7+	64.8	82.3+	83.2*&	**84.4***&$
[0.8, 0.9)	73.3	31.9	60.1+	61.9	74.2+	74.7*	**77.8***&$
[0.7, 0.8)	63.6	30.6	51.6+	51.4	62.9+	63.3	**66.1***&$
[0.6, 0.7)	43.6	29.0	39.9+	38.3	46.3+*	46.5*	**48.0***&$
[0.5, 0.6)	27.4	27.6	32.5+	28.9	34.5+*	34.9*&	**37.1***&$
[0.4, 0.5)	15.4	27.2	27.4	27.3	27.5	**27.8**	26.8
[0.3, 0.4)	8.2	**23.9**	22.7	23.8	22.4	22.4	22.4
(0.0, 0.3)	4.1	**24.6**	24.3	24.2	23.7	24.1	23.2
(0.0, 1.0)	40.2	28.3	40.6+	40.5	47.4+*	47.7*&	**48.8***&$

[a] "+" indicates that it significantly improves the translation results ("SMT+" vs. "SMT" and "Model-I" vs. "Model-I+"); "*" indicates that it is significantly better than both the TM and the SMT+ baselines; "&" means that it is significantly better than the Model-I+; and "$" means that it is significantly better than the Model-II.

combination to alleviate this problem as follows:

$$\begin{aligned}&\log P(TCM_k|STMI_k, SCM_k, NLN_k, CRL_k, SPL_k, SEP_k, z)\\&= \alpha \times \log P_{train}(TCM_k|SCM_k, NLN_k, CRL_k, SPL_k, SEP_k, STMI_k, z)\\&+(1-\alpha) \times \log P_{dev}(TCM_k|SCM_k, STMI_k, z)\end{aligned} \tag{8}$$

$$\begin{aligned}&\log P(STMI_k|SCM_k, NLN_k, z)\\&= \beta \times \log P_{train}(STMI_k|SCM_k, NLN_k, z)\\&+(1-\beta) \times \log P_{dev}(STMI_k|SCM_k, NLN_k, z)\end{aligned} \tag{9}$$

Where α and β are the weights (obtained from the development set) for the TM factors trained on the SMT training set and range from 0.0 to 1.0. Because there are only hundreds of sentence pairs in each interval of the development set, we drop several features in the corresponding factors of the development set. For example, we ignore NLN, CRL, SPL, and SEP in evaluating the TCM factor in the development set. This is the *Adapted Model-III*, and the results are shown in Table 12. Scores marked with "*" indicate that they are significantly better ($p < 0.05$) than both the TM and the SMT+, "@" means that they are significantly better ($p < 0.05$) than the Model-III, and "#" means that they are significantly better ($p < 0.05$) than Koehn-10. The bold entries are the best results in that interval.

It can be observed that the Adapted Model-III is significantly better than both the TM and the SMT+ systems when the fuzzy match score is above 0.5. In addition, the adapted approach significantly improves the translation results when the fuzzy match score is above 0.7 (Adapted Model-III vs. Model-III). Furthermore, the Adapted Model-III achieves the best BLEU score in most intervals and outperforms Koehn-10 in all intervals. The only exception is in the interval [0.6, 0.7), in which Koehn-10 is slightly better than the Adapted Model-III (48.5 vs. 48.3). However, all these results have still shown that the adapted approach is effective in this case.

In general, the proposed approach will work when the SMT can provide good enough output in the high-fuzzy-matching-ratio intervals (please see footnote #15 at page 21). Since the fuzzy matching ratio is measured against TM database, not against SMT training data, our approach is not sensitive to which corpus is adopted as the SMT training-set once the above condition is met. Adopting another TM database only changes the percentages of those high-fuzzy-matching-ratio intervals in the test set.

4.3.3. Proposed Model-IV (incorporating Top-N TM sentence-pairs)

The models derived above are all based on the most similar TM sentence pair. However, TM can provide help only when the given input source phrase can be matched by a corresponding TM phrase pair. If multiple relevant TM sentences are provided, then additional input source phrases could be covered by the union of those phrase pairs extracted separately from various TM sentences. Therefore, the original formulation $P(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}, tm\_s, tm\_t, tm\_f, s\_a, tm\_a)$ (Eq. (3), Section 2.1) is further replaced by $P(\bar{t}_1^K|\bar{s}_{a(1)}^{a(K)}, [tm\_s_i, tm\_t_i, tm\_f_i, s\_a_i, tm\_a_i]_{i=1}^N)$ to incorporate Top-N TM sentence pairs, where $[tm\_s_i, tm\_t_i, tm\_f_i, s\_a_i, tm\_a_i]_{i=1}^N$ denotes the associated information from the ith TM

Table 12
Adaptation translation results[a] (BLEU) of the Model-III when the TM database and the SMT training set are from different domains.

Intervals	TM	SMT	SMT+	Model-III	Adapted Model-III	Koehn-10	Ma-11-U
[0.9, 1.0)	81.3	30.9	64.7+	84.4*&$	**86.1***@#	81.5	44.2
[0.8, 0.9)	73.3	31.9	60.1+	77.8*&$	**80.3***@#	76.5*	46.3
[0.7, 0.8)	63.6	30.6	51.6+	66.1*&$	**67.2***@	67.1*@	42.5
[0.6, 0.7)	43.6	29.0	39.9+	48.0*&$	48.3*	**48.5***	37.7
[0.5, 0.6)	27.4	27.6	32.5+	37.1*&$	**37.1***#	35.3*	34.2
[0.4, 0.5)	15.4	27.2	**27.4**	26.8	26.9#	**25.1**	30.2
[0.3, 0.4)	8.2	**23.9**	22.7	22.4	22.4#	20.7	25.4
(0.0, 0.3)	4.1	**24.6**	24.3	23.2	23.2#	18.8	25.3
(0.0, 1.0)	40.2	28.3	40.6+	48.8*&$	**49.4***#	47.1*	36.2

[a] "*" indicates that it is significantly better than both the TM and the SMT+; "@" means that it is significantly better than the Model-III; and "#" means that it is significantly better than Koehn-10.

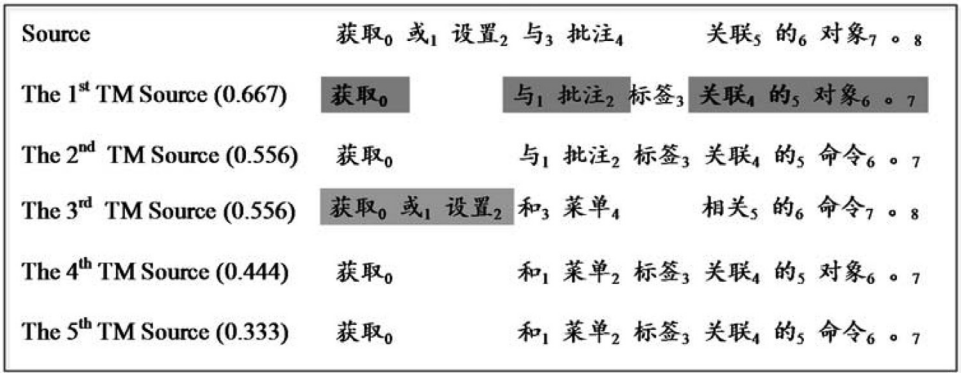


Fig. 7. The corresponding TM source phrase selection within Top-N TM sentences during decoding.

sentence-pair extracted from the TM database. As the result, the original meta/core factor $P(M_k|L_k, z)$ will be replaced by $P(M_{k,\,j(k)}|L_{k,\,j(k)}, z_{j(k)})$, where $j(k)(1 \le j(k) \le N)$ denotes that the corresponding TM phrase (for the target candidate $\bar{t}_k$ comes from the $j(k)$th TM sentence pair (please refer to Appendix C for detailed derivation). Eq. (10) shows our proposed Model-IV, which is modified from the Model-III by further considering that those matched TM phrase pairs might come from various TM sentences (in fact, the Model-III is just a special case of the Model-IV with N=1).

It is worth mentioning that the decoding procedure of this case is very similar to that for Eq. (4), except that it would check every TM sentence pair (within Top-N) instead of simply the Top-1 for each given source phrase (please refer to Appendix C for details). Furthermore, to distinguish those new phrase pairs from the original SMT phrase pairs, we also adopt the same strategy used in both the Model-II and the Model-III (i.e., adding ten additional feature weights).

$$
\begin{aligned}
&P(M_{k,j(k)}|L_{k,j(k)}, z_{j(k)}) \\
&\approx \left\{ \begin{array}{r}
P(TCM_{k,j(k)}|SCM_{k,j(k)}, NLN_{k,j(k)}, CRL_{k,j(k)}, SPL_{k,j(k)}, SEP_{k,j(k)}, z_{j(k)}) \\
\times P(CRL_{k,j(k)}|CSS_{k,j(k)}, SCM_{k,j(k)}, NLN_{k,j(k)}, SEP_{k,j(k)}, z_{j(k)}) \\
\times P(RPM_{k,j(k)}|TCM_{k,j(k)}, SCM_{k,j(k)}, NLN_{k,j(k)}, z_{j(k)}) \\
\times P(TPO_{k,j(k)}|SPO_{k,j(k)}, NLN_{k,j(k)}, z_{j(k)}) \\
\times P(STMI_{k,j(k)}|SCM_{k,j(k)}, NLN_{k,j(k)}, z_{j(k)})
\end{array} \right\}
\end{aligned} \tag{10}
$$

4.3.4. Experiments and results of Model-IV

Since TM can only give its hand when the given input source phrase is matched by a corresponding TM phrase pair, it is desirable to dynamically add all the matched phrase pairs into the SMT phrase table and then refer to them during decoding. However, because the TM information would not be reliable to guide the SMT decoding when the fuzzy match score is below 0.5 (according to the above mentioned experiments), we adopt the following rules to ensure the quality of the extracted TM information: (1) if the fuzzy match score of the first (Top-1) TM sentence is above 0.5, then only those TM sentence pairs whose fuzzy match scores are above 0.5 would be adopted to guide the SMT decoding. (2) If the fuzzy match score of the first TM sentence is below 0.5, then only those TM sentence pairs in the same FMS interval as the first TM sentence would be adopted. However, all the phrase-pairs from the Top-N TM sentences will be extracted.[12] Take Fig. 7 as an example, assume that we adopt the Top-5 TM sentence pairs, the fuzzy match score (FMS) of the best one is above 0.667, the FMS of the third one is above 0.5, and that the FMS of the fourth is below 0.5. Therefore, we would add all the matched phrase pairs extracted from the Top-5 TM sentences but only utilize those from the Top-3 to guide SMT decoding.

[12] We have also tried to add only the TM phrase-pairs extracted from the TM sentence pairs that meet the restrictions specified by those rules. However, the current version is slightly better as it can cover some additional unseen phrase-pairs.

Table 13
Translation results[a] (BLEU) of the Adapted Model-IV when the TM database and the SMT training set are from different domains.

Intervals	TM	SMT+	Adapted Model-IV					Koehn-10
			Top-1	Top-2	Top-5	Top-10	Top-15	
[0.9, 1.0)	81.3	64.7	86.1*#	87.0*#‡	88.0*#‡	**88.2***#	**88.2***#	81.5
[0.8, 0.9)	73.3	60.1	80.3*#	81.4*#‡	81.8*#	**82.3***#‡	**82.3***#	76.5*
[0.7, 0.8)	63.6	51.6	67.2*	69.5*#‡	69.8*#	**70.5***#‡	**70.5***#	67.1*
[0.6, 0.7)	43.6	39.9	48.3*	50.2*#‡	51.2*#‡	**51.3***#	**51.3***#	48.5*
[0.5, 0.6)	27.4	32.5	37.1*#	37.2*#	37.8*#‡	37.9 *#	**38.0***#	35.3*
[0.4, 0.5)	15.4	27.4	26.9#	27.1#	27.1#	27.4#	**27.6**#	25.1
[0.3, 0.4)	8.2	22.7	22.4#	23.1#‡	23.2#	**23.6**#	23.4#	20.7
(0.0, 0.3)	4.1	24.3	23.2#	23.4#	**23.5**#	**23.5**#	23.3#	18.8
(0.0, 1.0)	40.2	40.6	49.4*#	50.3*#‡	50.8*#‡	**51.1***#‡	**51.1***#	47.1*

[a] "*" indicates that it is significantly better than both the TM and the SMT+; "#" indicates that it is significantly better than Koehn10; and "‡" indicates that it is significantly better than the left adjacent Top-N (i.e., Top2 vs. Top1, Top5 vs. Top2, Top10 vs. Top5, and Top15 vs. Top10).

For each source phrase $\overline{s}_{a(k)}$in the input sentence, we scan from Top-1 to Top-M[13] ($M \leq N$) to find the corresponding TM source phrase and its associated target phrase pairs in each TM sentence pair according to the method described in Section 2.1. Therefore, we will have at most M corresponding TM source phrases $tm\_\overline{s}_{a(k)}$. For each given SMT source phrase, if several exactly matched TM source phrases $tm\_\overline{s}_{a(k)}$ could be found (i.e., SCM_k=Same), only the first TM sentence pair among them (i.e., with the highest FMS) would be adopted to guide the decoding process. In other words, all the TM-related features, such as TCM, SCM and so on would be extracted from this TM sentence pair. In contrast, if an exactly matched TM source phrase cannot be found, then we simply adopt the Top-1 TM sentence pair during decoding (because those partially matched TM phrase pairs are not reliable).

In addition, because we need $tm\_\overline{t}_{a(k-1),j(k)}$ to determine RPM_k, we should keep a record of the corresponding TM target phrase $tm\_\overline{t}_{a(k),j(k)}$ of the $j(k)$th TM sentence pair. That is, for each translation hypothesis, we would calculate the $P(M_{k,\,j(k)}|L_{k,\,j(k)}, z_{j(k)})$ M times (for the adopted Top-M TM sentence pairs) according to Eq. (10) and then keep the corresponding TM target phrase $tm\_\overline{t}_{a(k),j(k)}$ for each TM sentence pair. This is because the decoder may select any TM sentence pair among the Top-M candidates (as mentioned in the last paragraph).

For example, if the current source phrase $\overline{s}_{a(k)}$ is "获取$_0$" (gets), then there are three exactly matched TM source phrases that can be found within these Top-3 TM sentences; thus, we only select the first TM sentence to generate the corresponding TM features. In contrast, if the current source phrase $\overline{s}_{a(k)}$ is either"获取$_0$ 或$_1$ 设置$_2$" (gets or sets) or "或$_1$ 设置$_2$" (or sets), then only one exactly matched TM source phrase can be found from the third TM sentence. The third TM sentence is thus selected to generate the corresponding TM features. Finally, if the current source phrase $\overline{s}_{a(k)}$ is"设置$_2$ 与$_3$" (sets), then no exactly matched TM source phrase can be found within the three adopted TM sentences; thus, we simply use the first TM sentence to generate TM features. Afterwards, we select the best TM target phrase $tm\_\overline{t}_{a(k),j(k)}$ according to $P(M_{k,\,j(k)}|L_{k,\,j(k)}, z_{j(k)})$ with those generated features.

Table 13 presents the translation results of the Adapted[14] Model-IV with different Top-N TM sentence pairs. For reader's convenience, the results of the TM, the SMT+ and Koehn-10 are also provided in the tables for comparison. "*" indicates that it is significantly better than both the TM and the SMT+; "#" indicates that it is significantly better than Koehn-10; and "‡" indicates that it is significantly better than the left adjacent Top-N (i.e., Top-2 vs. Top-1, Top-5 vs. Top2, Top-10 vs. Top-5, and Top-15 vs. Top-10). It can be observed that adopting Top-N TM sentence pairs significantly improves the translation results of the Adapted Model-III (which is a special case of the Adapted Model-IV with N=1), especially for those high fuzzy match intervals. For example, in the interval [0.9, 1.0), using Top-2 TM sentence pairs, the Adapted Model-IV has been significantly improved from 86.1 to 87.0 (BLEU), and Top-5 further significantly increases the score to 88.0. In addition, the Adapted Model-IV with the Top-10 setting achieves the best translation results in most intervals based on the BLEU scores. However, only small improvements

[13] Because there is a restriction on the sentence fuzzy match score, we would only scan from Top-1 to Top-M ($M \leq N$). In the example of Fig. 8, we would scan from Top-1 to Top-3.

[14] Because the Adapted Model-IV (the same adaptation setting as we did for the Adapted Model-III) is much better than the Model-IV, we only incorporate Top-N TM sentence pairs to the adaptive version of the Model-IV.

can be obtained after N=10 because almost all reliable TM phrase pairs have been utilized by the Top-10 TM sentence pairs. In summary, with the help of Top-N ($N > 1$) TM sentence pairs, the Adapted Model-IV significantly outperforms the TM and the SMT+ systems in BLEU scores when the fuzzy match score is above 0.5. Additionally, the Adapted Model-IV ($N > 1$) significantly exceeds Koehn-10 in BLEU scores in all intervals. All these results illustrate that the proposed integrated models are effective and robust for the cross-domain tests.

Last, since incorporating Top-N TM sentence pairs actually is irrelevant to which model is adopted, it would be interesting to see if we can get the similar improvement from other models. We thus test the Model-I when the TM database and the SMT training set are the same (Table 4), and (the Model-II when the TM database and the SMT training set are different but from the same domain (Table 9). However, our experiments show that adopting Top-N TM sentence pairs can only obtain slight or even no improvement at various FMS intervals in those cases. This is mainly due to the following reason: In comparison with the SMT, those non-top-1 TM sentences are not considerable better when both the SMT and the TM are from the same domain; therefore, they cannot provide additional help. For example, in Table 1, SMT/SMT+ gives much better performance for these two cases in the first two FMS intervals (i.e., [0.9, 1.0) and [0.8, 0.9)); and they significantly outperform the TM of their next FMS intervals (i.e., [0.8, 0.9) and [0.7, 0.8) respectively). Since the corresponding FMS scores of the Top-2 TM candidates frequently degrade one to three FMS intervals when FMS is high (e.g., Top-1 TM is in [0.9, 1.0), but Top-2 TM is in [0.6, 0.7)), the corresponding target-phrases associated with those non-top-1 TM sentences are even inferior to those directly generated by the SMT/SMT+. For instance, in the interval [0.9, 1.0), the SMT is 81.4 (in BLEU score) in Table 4 and the SMT + is 73.6 in Table 9; nonetheless, the TM in the interval [0.8, 0.9) are only 73.3 and 72.7 in Table 4 and Table 10, respectively, in those cases. On the contrary, Table 1 shows that the SMT+ is 64.7 (at [0.9, 1.0)) and the TM is 73.3 (at [0.8, 0.9)) in the similar situation. In general, adopting more TM sentence pairs can be helpful only if the quality of the matched target phrases of those additional TM sentence pairs significantly outperform that of SMT (e.g., 73.3 vs. 64.7)

5. Related work

According to the classification made in Section 1, the previous studies of combining SMT and TM can be grouped into three categories. The first category uses a classifier (or a re-ranker) to judge whether TM or SMT provides a better translation sentence and then delivers the better translation to the post-editors (He et al., 2010a; 2010b; Dara et al., 2013). He et al. (2010a) first proposed a translation recommendation framework in which an SVM classifier is used to decide which method (the SMT translation or the TM translation) is more suitable for the post-editor. In addition, the recommendation confidence is also estimated by the SVM classifier. This approach aims to provide more suitable translation candidates among the TM and the SMT for human translators. Subsequently, He et al. (2010b) recast the recommendation framework as a ranking problem with N-best outputs from the SMT and K-best outputs from TM. In addition, (Dara et al., 2013) introduced the outputs of online translation systems such as the Google and Bing translators. Because the SMT and the TM outputs are not merged but simply re-ranked for post-editors, the possible improvements resulting from those approaches are quite limited.

The second category, which directly merges TM matched sub-segments into the SMT input sentence, translates the sentence in a pipelined manner. All those approaches first determine whether the extracted TM sentence pair should be adopted or not in the first stage. Most of these studies Koehn and Senellart (2010), Zhechev and Genabith (2010) used fuzzy match score as the threshold, but Ma et al. (2011a), Ma et al. (2011b) used a classifier to make the judgment. Subsequently, they merged the relevant translations of matched segments into the input sentence and then force the SMT system to only translate those unmatched sub-segments at decoding. Most of these approaches simply utilize word alignments to align TM phrases, but Zhechev and Genabith (2010) also incorporated syntax information for aligning TM phrases. Because of the drawbacks mentioned in Section 1, the approaches of this category still cannot obtain satisfactory results.

In the last category, the longest matched TM phrase pairs are added to the SMT phrase table (Biçici and Dymetman, 2008; Simard and Isabelle, 2009), and they are associated with a fixed large probability value to favor the TM target phrase during SMT decoding. However, there is a difference between Biçici and Dymetman (2008) and Simard and Isabelle (2009): the former allows their phrases to be noncontiguous (i.e., to have explicit gaps). In addition, Simard and Isabelle (2009) incorporated the TM information as features for re-ranking the SMT outputs but only achieved a slight improvement. Because only one aligned target phrase will be added for each matched source

phrase in these approaches, they share most of the drawbacks from the pipeline approaches of the second category. Furthermore, Koehn and Senellart (2010) combined the TM with a hierarchical phrase-based SMT system. They constructed the hierarchical phrase rules by keeping the matched parts and replacing the unmatched parts with non-terminals. This method also achieved significant improvement when the fuzzy match score is above 0.7.

In addition to the above mentioned approaches, some earlier work also tried to combine the TM and the SMT systems. Among them, Marcu (2001) selected contiguous alignments to construct a statistical translation memory. He subsequently adopted this statistical translation memory in a word-based decoder and obtained better translations. However, this method was regarded as an early manifestation of the phrase-based translation model by Biçici and Dymetman (2008). In addition, Hewavitharana et al. (2005) identified the mismatch between the source sentence and the best fuzzy match. Subsequently, the SMT system is used to translate the mismatch and modify the translation and obtain the desired results. In their implementation, an XML input is not constructed.

Besides, our ACL and COLING papers (Wang et al., 2013; 2014) also integrated TM information and TM matched phrase pairs into the SMT system. However, the ACL version only focuses on the case where the TM database and the SMT training set are the same, and the COLING paper merely dynamically adds those matched TM phrase pairs into the SMT phrase table. Both only utilize the Top-1 TM sentence pair, not the Top-N TM sentence pairs. Moreover, Li et al. (2014, 2016) had incorporated all the features proposed in our Model-I into a discriminative log-linear framework. However, they only achieved the comparable results for our first scenario (in which SMT and TM share the dataset), and they do not consider other scenarios. Furthermore, because the example-based machine translation (EBMT Nagao, 1984) model is a similar approach to that of using TM, some approaches combine EBMT (instead of TM) with SMT. Watanabe and Sumita (2003) adopted similar examples generated by the EBMT system as the initial translations and subsequently used a word-based SMT system to revise the translations. Smith and Clark (2009) utilized the EBMT system to extract the best fuzzy match and subsequently used the XML markup method to translate the unmatched parts. Phillips (2011) adopted some SMT features into the EBMT system. Dandapat et al. (2011, 2012) incorporated the SMT phrase table as an additional example database for EBMT. Since different data sets were adopted, it is difficult to directly compare their performances with that of ours.

On the other hand, some approaches adopted document related features (not sentence related features as that proposed in this paper) to improve the translation performance. For example, Snover et al. (2008) proposed to first search monolingual documents in the target language that might be comparable to the source documents, and then improve the SMT performance via utilizing the monolingual data included in those target documents. Also, inspired by the "one translation per discourse" observation, Ture et al. (2012) added three document-scale features to the translation model to enhance the translation consistency in the same document. Besides, Eidelman et al. (2012) proposed to use topic distributions to compute the topic-dependent lexical weighting probabilities and directly incorporate them into the translation model as features. Although those document-level features mentioned above could improve the performance, a human-generated target sentence (i.e., TM sentence-pair) provides more sentence-specific local context information. In comparison with them, our models utilize sentence-level target information and reflect the local context more closely.

Furthermore, some approaches improved the translation performance by integrating the source side contextual features to supplement the traditional PB-SMT framework which only considers the co-occurrence frequency of source and target phrases. Weller et al. (2010) proposed a method which estimates the "contextually conditioned translation probability" to resolve the ambiguities by separating the entire set of translation candidates into various subsets for different situations. Haque et al. (2009, 2010, 2011) used a range of contextual features, including lexical features of neighbouring words, supertags, dependency information, and semantic roles. They observed that including contextual features of the source sentence in general produces improvements. Basically, the nature of these works is feature engineering under the classical PB-SMT framework.

Unlike those previous approaches, we propose a unified approach that incorporates the TM information of each source phrase into the phrase-based SMT during decoding under an integrated model, and are the first to approach this problem in a principled way. Those integrated models jointly consider the probability information of both the SMT and TM factors. In addition, all possible TM target phrases are kept, and the proposed models select the best phrase by referring to both the SMT and the TM information. Furthermore, to improve the coverage of the unseen

phrase pairs, we dynamically merge the TM phrase pairs into the SMT phrase table when the TM database and the SMT training set are different. Lastly, to increase the coverage rate of matched source phrases, we adopt Top-N TM sentence pairs during the decoding process.

6. Conclusion and future work

This paper proposes a unified approach to integrate translation memory into phrase-based machine translation in a principled way. Unlike previous two-stage pipelining approaches, which directly merge TM results into the input sentences at the surface level, the proposed framework refers to the corresponding TM information associated with each target phrase during SMT decoding at a deep level. Under this unified framework, several integrated models are proposed to incorporate different types of information extracted from TM to guide SMT decoding. Because we might have multiple aligned TM target phrases for a given TM source phrase, the proposed models consider all the possible candidates and select the best one during decoding. In addition, the SMT phrase table is dynamically enhanced with these new TM phrase pairs when the TM database and the SMT training set are different. Furthermore, the Top-N TM sentence pairs are incorporated into the integrated model to increase the coverage of those matched TM source phrases when the TM database and the SMT training set are from different domains.

Various experiments have been conducted on a ChineseEnglish computer technical document TM database. When the TM database and the SMT training set are the same, our experiments show that the proposed Model-I (which utilizes TM phrase-pairs and their relative positions) significantly improves the translation quality over both the phrase-based SMT system and the TM system when the fuzzy match score is above 0.4. Next, when the TM database and the SMT training set are different but from the same domain, our experiments show that dynamically merging TM phrase pairs into the SMT phrase table considerably improves the translation quality of the Model-I. Also, the proposed Model-II (which additionally distinguishes the newly added TM phrase-pairs from those original SMT phrase pairs) is significantly better than both the SMT and the TM systems when the fuzzy match score is above 0.4. Last, when the TM database and the SMT training set are from different domains, the proposed Model-III (which also determines if there is an exactly matched TM target phrase candidate for the given SMT target phrase) performs significantly better compared to both the TM and the SMT systems. Besides, with the help of the Top-N relevant TM sentences, the Adapted Model-IV further improves the translation results for cross-domain tests. In summary, in all test conditions, the proposed models significantly outperform the state-of-the-art approaches. We believe that the improvement is mainly due to that the proposed approach can allow the SMT utilize global context with a local dependency model.

It is also worth to mention that the proposed models do not utilize any language-dependent features, linguistic rules or tree banks, although the experiments are conducted on ChineseEnglish language pair. Therefore, it is expected that the proposed approach can be applied to other language pairs with little (or no) adaptation effort.

Finally, because the tree-based translation models (Galley et al., 2004; 2006; Huang et al., 2006; Liu et al., 2006; 2007; 2009; Mi and Huang, 2008; Chiang, 2010; Xiao and Zhu, 2013) are drawing increased attention in the community of statistical machine translation, it would also be interesting to integrate translation memory into the tree-based translation models. On the other hand, Neural Machine Translation (NMT) (Bahdanau et al., 2014; Sutskever et al., 2014) has shown great success recently. Since the current NMT system does not utilize the human-generated sentence-specific target information (which reflects the local context more closely) that could be provided by the TM sentence-pair, it is expected that the performance of NMT will be further improved if TM information could be jointly considered, and it also would be our future work.

Acknowledgment

The authors would like to thank Dr. Yanjun Ma and Dr. Yifan He for discussing their works during this study.

Appendix A. The derivation of adopted translation formulation

As we would like to integrate the TM into the phrase-based SMT framework, the sentences should be converted into their associated phrase sequences. Therefore, $P(t|s, tm\_s, tm\_t, tm\_f, s\_a, tm\_a)$ in Eq. (2) is derived as:

$$\begin{aligned} & P(t|s, [tm\_s, tm\_t, tm\_f, s\_a, tm\_a]) \\ & = \sum_{[\bar{s}_1^K = s, \bar{t}_1^K = t]} P(\bar{t}_1^K | \bar{s}_{a(1)}^{a(K)}, [tm\_s, tm\_t, tm\_f, s\_a, tm\_a]) \\ & \approx max_{[\bar{s}_1^K = s, \bar{t}_1^K = t]} P(\bar{t}_1^K, \bar{s}_{a(1)}^{a(K)} | s, tm\_s, tm\_t, tm\_f, s\_a, tm\_a) \\ & = max_{[\bar{s}_1^K = s, \bar{t}_1^K = t]} \begin{pmatrix} P(\bar{t}_1^K | \bar{s}_{a(1)}^{a(K)}, tm\_s, tm\_t, tm\_f, s\_a, tm\_a) \\ \times P(\bar{s}_1^K | s, tm\_s, tm\_t, tm\_f, s\_a, tm\_a) \end{pmatrix} \end{aligned} \tag{A.1}$$

Where $\bar{s}_{a(k)}$ and $\bar{t}_k$ denote the kth associated source phrase and target phrase, respectively. Additionally, $\bar{s}_{a(1)}^{a(K)}$ and $\bar{t}_1^K$ denote the associated source phrase sequence and the target phrase sequence, respectively (suppose that there are a total of K phrases without phrase insertions). If $\bar{s}_{a(k)}$ is a deleted phrase, then the corresponding $\bar{t}_k$ would be "ϕ". In the last line of Eq. (A.1), we first segment the given source sentence into various phrases and then translate the sentence based on those source phrases. In addition, $\bar{s}_{a(1)}^{a(K)}$ is replaced by $\bar{s}_1^K$ in the second probability factor, as they are actually the same segmentation sequence. Since the source phrase segmentation of s is mostly irrelevant to TM information, the second factor of Eq. (A.1) can be simplified as below:

$$P(\bar{s}_1^K | s, tm\_s, tm\_t, tm\_f, s\_a, tm\_a) \approx P(\bar{s}_1^K | s) \tag{A.2}$$

Since this factor is found to have negligible effect on improving the translation performance in our experiments, it can be ignored, and we only need to consider the first factor $P(\bar{t}_1^K | \bar{s}_{a(1)}^{a(K)}, tm\_s, tm\_t, tm\_f, s\_a, tm\_a)$.

Appendix B. Target phrase adjacent candidate relative position matching status (RPM)

Recalling that $\bar{t}_k$ is always right adjacent to $\bar{t}_{k-1}$, the corresponding values of RPM_s for various cases are then specified as follows:

(1) If both $tm\_\bar{t}_{a(k-1)}$ and $tm\_\bar{t}_{a(k)}$ are not null:

  (a) if $tm\_\bar{t}_{a(k)}$ is on the right of $tm\_\bar{t}_{a(k-1)}$ and they are also adjacent to each other:

    (i) If the right boundary words of $\bar{t}_{k-1}$ and $tm\_\bar{t}_{a(k-1)}$ are the same and the left boundary words of $\bar{t}_k$ and $tm\_\bar{t}_{a(k)}$ are also the same, then the associated RPM_k is "Adjacent-Same";
    (ii) Otherwise, RPM_k is "Adjacent-Substitute";

  (b) If $tm\_\bar{t}_{a(k)}$ is on the right of $tm\_\bar{t}_{a(k-1)}$ but they are not adjacent to each other, the current RPM_k is "*Linked−Interleaved*";
  (c) If $tm\_\bar{t}_{a(k)}$ is not on the right of $tm\_\bar{t}_{a(k-1)}$:

    (i) If there are cross parts between $tm\_\bar{t}_{a(k)}$ and $tm\_\bar{t}_{a(k-1)}$, then the associated RPM_k is "Linked-Cross";
    (ii) Otherwise, RPM_k is "Linked-Reversed";

(2) If $tm\_\bar{t}_{a(k-1)}$ is null but $tm\_\bar{t}_{a(k)}$ is not null, then find the first $tm\_\bar{t}_{a(k-n)}(k \geq n)$ that is not null (n starts from 2):[15]

  (a) If $tm\_\bar{t}_{a(k)}$ is on the right of $tm\_\bar{t}_{a(k-n)}$, then the associated RPM_k is "*Skip−Forward*";
  (b) If $tm\_\bar{t}_{a(k)}$ is not on the right of $tm\_\bar{t}_{a(k-n)}$;

[15] It can be identified by simply memorizing the index of the nearest non-null $tm\_\overline{a(k-n)}$ during search.

(i) If there are cross parts between $tm\_\bar{t}_{a(k)}$ and $tm\_\bar{t}_{a(k-n)}$, then the associated RPM_k is "*Skip−Cross*";
(ii) Otherwise, RPM_k is "*Skip−Reversed*".

(3) If $tm\_\bar{t}_{a(k)}$ is null, then the associated RPM_k is "*NA*".

In Fig. 1, assume that $\bar{t}_{k-1}, \bar{t}_k$ and $tm\_\bar{t}_{a(k-1)}$ are "gets an", "object that is *associated with*" and "gets$_0$ an$_1$", respectively. For the $tm\_\bar{t}_{a(k)}$ "object$_2$ that$_3$ is$_4$ associated$_5$", because $tm\_\bar{t}_{a(k)}$ is on the right of $tm\_\bar{t}_{a(k-1)}$, they are adjacent pair, and both boundary words ("an" and "an$_1$"; "object" and "object$_2$") are matched; therefore, RPM_k is "Adjacent-Same". For the $tm\_\bar{t}_{a(k)}$ "an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$", because there is a cross part "an$_1$" between $tm\_\bar{t}_{a(k)}$ and $tm\_\bar{t}_{a(k-1)}$, RPM_k is "*Linked-Cross*". In contrast, assume that $\bar{t}_{k-1}, \bar{t}_k$ and $tm\_\bar{t}_{a(k-1)}$ are "gets", "object that is associated with" and "gets$_0$", respectively. For the $tm\_\bar{t}_{a(k)}$ "an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$ ", because $tm\_\bar{t}_{a(k)}$ and $tm\_\bar{t}_{a(k-1)}$ are adjacent pair and because the left boundary words of $\bar{t}_k$ and $tm\_\bar{t}_{a(k)}$ (i.e., "object" and "an$_1$") are not matched, RPM_k is "Adjacent-Substitute". For the $tm\_\bar{t}_{a(k)}$ "object$_2$ that$_3$ is$_4$ associated$_5$", because $tm\_\bar{t}_{a(k)}$ is on the right of $tm\_\bar{t}_{a(k-1)}$ and because they are not adjacent pair, RPM_k is "Linked-Interleaved". As the last example, assume that $\bar{t}_{k-1}, \bar{t}_k$ and $tm\_\bar{t}_{a(k-1)}$ are "the annotation label", "object that is associated with" and "the$_7$ annotation$_8$ label$_9$", respectively. For the $tm\_\bar{t}_{a(k)}$ "an$_1$ object$_2$ that$_3$ is$_4$ associated$_5$", because $tm\_\bar{t}_{a(k)}$ is on the left of $tm\_\bar{t}_{a(k-1)}$ and because there are no cross parts, RPM_k is "Linked-Reversed".

Appendix C. The derivation of translation formula for Top-N case

Eq. (4) describes the framework that incorporates only Top-1 TM sentence pair. When Top-N sentence pair are simultaneously adopted, Eq. (4) should be re-formulated as follows:

$$
\begin{aligned}
& P(\bar{t}_1^K | \bar{s}_{a(1)}^{a(K)}, [tm\_s_i, tm\_t_i, tm\_f_i, s\_a_i, tm\_a_i]_{i=1}^N) \\
&= \textstyle\sum_{tm\_\bar{t}_{a(1),j(1)}^{a(K),j(K)}} P\left(\bar{t}_1^K, tm\_\bar{t}_{a(1),j(1)}^{a(K),j(K)} | \bar{s}_{a(1)}^{a(K)}, [tm\_\bar{s}_{a(1),i}^{a(K),i}, tm\_t_i, z_i]_{i=1}^N\right) \\
&\approx max_{tm\_\bar{t}_{a(1),j(1)}^{a(K),j(K)}} P\left(\bar{t}_1^K, tm\_\bar{t}_{a(1),j(1)}^{a(K),j(K)} | \bar{s}_{a(1)}^{a(K)}, [tm\_\bar{s}_{a(1),i}^{a(K),i}, tm\_t_i, z_i]_{i=1}^N\right) \\
&\approx max_{tm\_\bar{t}_{a(1),j(1)}^{a(K),j(K)}} P\left(\bar{t}_1^K, M_{1,j(1)}^{K,j(K)} | \bar{s}_{a(1)}^{a(K)}, L_{1,j(1)}^{K,j(K)}, tm\_t_{j(1)}^{j(K)}, z_{j(1)}^{j(K)}\right) \\
&\approx max_{tm\_\bar{t}_{a(1),j(1)}^{a(K),j(K)}} \left\{P\left(\bar{t}_1^K | \bar{s}_{a(1)}^{a(K)}\right) \times P\left(M_{1,j(1)}^{K,j(K)} |, L_{1,j(1)}^{K,j(K)}, tm\_t_{j(1)}^{j(K)}, z_{j(1)}^{j(K)}\right)\right\} \\
&\approx P\left(\bar{t}_1^K | \bar{s}_{a(1)}^{a(K)}\right) \times \prod_{k=1}^{K} \max_{tm\_\bar{t}_{a(k),j(k)}} P(M_{k,j(k)} | L_{k,j(k)}, z_{j(k)}) / C_k^{'}
\end{aligned}
\tag{C.1}
$$

Where $j(k)(1 \le j(k) \le N)$ denotes that the corresponding TM phrase comes from the $j(k)$th TM sentence pair. For example, $tm\_\bar{t}_{a(k),(k)}$ means that the current $tm\_\bar{t}_{a(k)}$ is from the $j(k)$th TM target sentence $tm\_t_{j(k)}$. Similarly, $M_{k,\,j(k)}$, $L_{k,\,j(k)}$ and $z_{j(k)}$ are the corresponding TM phrase matching status (for the target candidate $\bar{t}_k$), the matching statusvector (of $\bar{s}_{a(k)}$), and the TM fuzzy match interval in the $j(k)$th TM sentence pair, respectively. Finally, $C_K^{'} = \sum_{tm\_\bar{t}_{a(k)j(k)}} P(M_{k,j(k)} | L_{k,j(k)}, z_{j(k)})$ is a corresponding normalization value,[16] which will be ignored during decoding.

In adopting the above equation, since evaluating RPM_k of $[\bar{t}_{k-1}, \bar{t}_k]$ would be meaningless if $tm\_\bar{t}_{a(k-1)}$ and $tm\_\bar{t}_{a(k)}$ come from two different TM sentence pairs, $[tm\_\bar{t}_{a(k-1)}, tm\_\bar{t}_{a(k)}]$ would be replaced by $[tm\_\bar{t}_{a(k-1),j(k)}, tm\_\bar{t}_{a(k),j(k)}]$. In other words, $tm\_\bar{t}_{a(k-1),j(k-1)}$ should be replaced by $tm\_\bar{t}_{a(k-1),j(k)}$ while evaluating RPM_k if multiple TM relevant sentences are adopted. For other features associated with $\bar{t}_k$, they should all be evaluated using the same TM sentence pair (indexed by $j(k)$ within the Top-N) that not only exactly matches $\bar{s}_{(}a(k)_{)}$ (with SCM=Same) but also possesses the highest FMS among them. If there is no sentence pair that can exactly match $\bar{s}_{a(k)}$, the Top-1 TM sentence would be adopted to generate features for $\bar{t}_k$.

[16] The summation will be taken over various $tm\_\bar{t}_{a(k),j(k)}$ that are associated with $tm\_t_{j(k)}$.

References

Auli, M., Lopez, A., Hoang, H., Koehn, P., 2009. A systematic analysis of translation model search spaces. In: Proceedings of the Fourth Workshop on Statistical Machine Translation. Association for Computational Linguistics, pp. 224–232.

Bahdanau, D., Cho, K., Bengio, Y., 2014. Neural Machine Translation by Jointly Learning to Align and Translate. arXiv:1409.0473.

Barrachina, S., Bender, O., Casacuberta, F., Civera, J., Cubel, E., Khadivi, S., Lagarda, A., Ney, H., Tomás, J., Vidal, E., et al., 2009. Statistical approaches to computer-assisted translation. Comput. Linguist. 35 (1), 3–28.

Bell, T.C., Cleary, J.G., Witten, I.H., 1990. Text Compression. Prentice-Hall, Inc.

Biçici, E., Dymetman, M., 2008. Dynamic translation memory: using statistical machine translation to improve translation memory fuzzy matches. Comput. Linguist. Intell. Text Process. 4919, 454–465.

Cancedda, N., 2012. Private access to phrase tables for statistical machine translation. In: Proceedings of the Fiftieth Annual Meeting of the Association for Computational Linguistics: Short Papers, 2. Association for Computational Linguistics, pp. 23–27.

Chen, S.F., Goodman, J., 1996. An empirical study of smoothing techniques for language modeling. In: Proceedings of the Thirty-Fourth Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 310–318.

Cherry, C., 2013. Improved reordering for phrase-based translation using sparse features. In: Proceedings of the 2013 HLT-NAACL, pp. 22–31.

Chiang, D., 2005. A hierarchical phrase-based model for statistical machine translation. In: Proceedings of the Forty-Third Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 263–270.

Chiang, D., 2010. Learning to translate with source and target syntax. In: Proceedings of the Forty-Eighth Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 1443–1452.

Chiang, D., Marton, Y., Resnik, P., 2008. Online large-margin training of syntactic and structural translation features. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 224–233.

Dandapat, S., Morrissey, S., Way, A., Forcada, M.L., 2011. Using example-based MT to support statistical MT when translating homogeneous data in a resource-poor setting. In: Proceedings of the Fifteenth Annual Meeting of the European Association for Machine Translation (EAMT 2011), pp. 201–208.

Dandapat, S., Morrissey, S., Way, A., Van Genabith, J., 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). Association for Computational Linguistics, pp. 48–58.

Dara, A.A., Dandapat, S., Groves, D., Van Genabith, J., 2013. TMTprime: a recommender system for MT and TM integration. In: Proceedings of the 2013 HLT-NAACL, pp. 10–13.

Eidelman, V., Boyd-Graber, J., Resnik, P., 2012. Topic models for dynamic translation model adaptation. In: Proceedings of the Fiftieth Annual Meeting of the Association for Computational Linguistics: Short Papers, 2. Association for Computational Linguistics, pp. 115–119.

Galley, M., Graehl, J., Knight, K., Marcu, D., DeNeefe, S., Wang, W., Thayer, I., 2006. Scalable inference and training of context-rich syntactic translation models. In: Proceedings of the Twenty-First International Conference on Computational Linguistics and the Forty-Fourth Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 961–968.

Galley, M., Hopkins, M., Knight, K., Marcu, D., 2004. What's in a Translation Rule. Technical report. Columbia University: Department of Computer Science, New York.

Galley, M., Manning, C.D., 2008. A simple and effective hierarchical phrase reordering model. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 848–856.

Galley, M., Quirk, C., Cherry, C., Toutanova, K., 2013. Regularized minimum error rate training. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 1948–1959.

Haque, R., Naskar, S.K., Van Den Bosch, A., Way, A., 2010. Supertags as source language context in hierarchical phrase-based SMT. Association for Machine Translation in the Americas.

Haque, R., Naskar, S.K., van den Bosch, A., Way, A., 2011. Integrating source-language context into phrase-based statistical machine translation. Mach. Transl. 25 (3), 239–285.

Haque, R., Naskar, S. K., Ma, Y., Way, A., 2009. Using Supertags as Source Language Context in SMT. In: EAMT 2009 - 13th Annual Conference of the European Association for Machine Translation, 13-15 May 2009, Barcelona, Spain.

He, Y., Ma, Y., van Genabith, J., Way, A., 2010. Bridging SMT and TM with translation recommendation. In: Proceedings of the Forty-Eighth Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 622–630.

He, Y., Ma, Y., Way, A., Van Genabith, J., 2010. Integrating N-best SMT outputs into a TM system. In: Proceedings of the Twenty-Third International Conference on Computational Linguistics: Posters. Association for Computational Linguistics, pp. 374–382.

Hewavitharana, S., Vogel, S., Waibel, A., 2005. Augmenting a statistical translation system with a translation memory. In: Proceedings of the 2005 European Association. for Machine Translation, EAMT, 5.

Hopkins, M., May, J., 2011. Tuning as ranking. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 1352–1362.

Huang, L., Chiang, D., 2007. Forest rescoring: faster decoding with integrated language models. In: Proceedings of the 2007 Annual Meeting of the Association For Computational Linguistics, 45, p. 144.

Huang, L., Knight, K., Joshi, A., 2006. A syntax-directed translator with extended domain of locality. In: Proceedings of the 2006 Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing. Association for Computational Linguistics, pp. 1–8.

Kneser, R., Ney, H., 1995. Improved backing-off for m-gram language modeling. In: Proceedings of the 1995 International Conference on Acoustics, Speech, and Signal Processing, ICASSP-95, 1. IEEE, pp. 181–184.

Koehn, P., 2004. Statistical significance tests for machine translation evaluation. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 388–395.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., et al., 2007. Moses: open source toolkit for statistical machine translation. In: Proceedings of the Forty-Fifth Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions. Association for Computational Linguistics, pp. 177–180.

Koehn, P., Och, F.J., Marcu, D., 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, 1. Association for Computational Linguistics, pp. 48–54.

Koehn, P., Senellart, J., 2010. Convergence of translation memory and statistical machine translation. In: Proceedings of the 2010 AMTA Workshop on MT Research and the Translation Industry, pp. 21–31.

Lagoudaki, E., 2006. Translation memories survey 2006: user perceptions around TM use. proceedings of the 2006 ASLIB International Conference Translating the Computer, 28, pp. 1–29.

Li, L., Escartín, C.P., Way, A., Liu, Q., 2016. Combining translation memories and statistical machine translation using sparse features. Mach. Transl. 30 (3−4), 183–202.

Li, L., Way, A., Liu, Q., 2014. A discriminative framework of integrating translation memory features into SMT. In: Proceedings of the Eleventh Conference of the Association for Machine Translation in the Americas, 1, pp. 249–260.

Li, Q., Juang, B.-H., Lee, C.-H., 2000. Automatic verbal information verification for user authentication. IEEE Trans. Speech Audio Process. 8 (5), 585–596.

Liu, Q., Zhao, H., 2009. Report on CWMT2009 MT translation evaluation. In: Proceedings of the Fifth China Workshop on Machine Translation (CWMT2009), pp. 1–31.

Liu, Y., Huang, Y., Liu, Q., Lin, S., 2007. Forest-to-string statistical translation rules. In: Proceedings of the 2007 Association for Computational Linguistics, ACL, pp. 704–711.

Liu, Y., Liu, Q., Lin, S., 2006. Tree-to-string alignment template for statistical machine translation. In: Proceedings of the Twenty-First International Conference on Computational Linguistics and the Forty-Fourth Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 609–616.

Liu, Y., Lü, Y., Liu, Q., 2009. Improving tree-to-tree translation with packed forests. In: Proceedings of the Joint Conference of the Forty-Seventh Annual Meeting of the ACL and the Fourth International Joint Conference on Natural Language Processing of the AFNLP, 2. Association for Computational Linguistics, pp. 558–566.

Ma, Y., He, Y., Way, A., van Genabith, J., 2011a. Consistent translation using discriminative learning: a translation memory-inspired approach. In: Proceedings of the Forty-Ninth Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 1. Association for Computational Linguistics, pp. 1239–1248.

Ma, Y.H.Y., Way, A., van Genabith, J., 2011b. Rich Linguistic Features for Translation Memory-Inspired Consistent Translation. In: Proceedings of the Thirteenth Machine Translation Summit, pp. 456–463.

Marcu, D., 2001. Towards a unified approach to memory-and statistical-based machine translation. In: Proceedings of the Thirty-Ninth Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 386–393.

Mi, H., Huang, L., 2008. Forest-based translation rule extraction. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 206–214.

Nagao, M., 1984. A framework of a mechanical translation between Japanese and English by analogy principle. Artif. Hum. Intell. 2938, 351–354.

Och, F.J., 2003. Minimum error rate training in statistical machine translation. In: Proceedings of the Forty-First Annual Meeting on Association for Computational Linguistics, 1. Association for Computational Linguistics, pp. 160–167.

Och, F.J., Ney, H., 2002. Discriminative training and maximum entropy models for statistical machine translation. In: Proceedings of the Fortieth Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 295–302.

Papineni, K., Roukos, S., Ward, T., Zhu, W.-J., 2002. BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the Fortieth Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 311–318.

Phillips, A.B., 2011. Cunei: open-source machine translation with relevance-based models of each translation instance. Mach. Transl. 25 (2), 161–177.

Sikes, R., 2007. Fuzzy matching in theory and practice. Multilingual 18 (6), 39–43.

Simard, M., Isabelle, P., 2009. Phrase-based machine translation in a computer-assisted translation environment. In: Proceedings of the Twelfth Machine Translation Summit (MT Summit XII), pp. 120–127.

Smith, J., Clark, S., 2009. EBMT for SMT: a new EBMT-SMT hybrid. In: Proceedings of the Third International Workshop on Example-Based Machine Translation, pp. 3–10.

Snover, M., Dorr, B., Schwartz, R., 2008. Language and translation model adaptation using comparable corpora. In: Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 857–866.

Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J., 2006. A study of translation edit rate with targeted human annotation. In: Proceedings of the 2006 Association for Machine Translation in the Americas, 200.

Stolcke, A., et al., 2002. SRILM − an extensible language modeling toolkit. In: Proceedings of the 2002 Interspeech, 2002, p. 2002.

Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. In: Proceedings of the 2014 Advances in Neural Information Processing Systems, pp. 3104–3112.

Ture, F., Oard, D.W., Resnik, P., 2012. Encouraging consistent translation choices. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, pp. 417–426.

Wang, K., Zong, C., Su, K.-Y., 2013. Integrating translation memory into phrase-based machine translation during decoding. In: Proceedings of the Fifty-First Annual Meeting of the Association for Computational Linguistics (Long Papers), 1, pp. 11–21.

Wang, K., Zong, C., Su, K.-Y., et al., 2014. Dynamically integrating cross-domain translation memory into phrase-based machine translation during decoding. In: Proceedings of the 2014 COLING, pp. 398–408.

Watanabe, T., Sumita, E., 2003. Example-based decoding for statistical machine translation. In: Proceedings of the 2003 Machine Translation Summit IX, pp. 410–417.

Watanabe, T., Suzuki, J., Tsukada, H., Isozaki, H., 2007. Online large-margin training for statistical machine translation. In: Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing, EMNLP. Citeseer.

Weller, M., et al., 2010. An Empirical Analysis of Source Context Features for Phrase-Based Statistical Machine Translation. Universität Stuttgart Ph.D. thesis, Diploma thesis.

Wisniewski, G., Allauzen, A., Yvon, F., 2010. Assessing phrase-based translation models with oracle decoding. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 933–943.

Witten, I.H., Bell, T.C., 1991. The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression. IEEE Trans. Inf. Theory 37 (4), 1085–1094.

Xiao, T., Zhu, J., 2013. Unsupervised sub-tree alignment for tree-to-tree translation. J. Artif. Intell. Res. 48, 733–782.

Yamada, K., Knight, K., 2001. A syntax-based statistical translation model. In: Proceedings of the Thirty-Ninth Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 523–530.

Zhechev, V., Genabith, J., 2010. Seeding statistical machine translation with translation memory output through tree-based structural alignment. In: Proceedings of the Fourth Workshop on Syntax and Structure in Statistical Translation, pp. 43–51.

Zollmann, A., Venugopal, A., Och, F., Ponte, J., 2008. A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In: Proceedings of the Twenty-Second International Conference on Computational Linguistics, 1. Association for Computational Linguistics, pp. 1145–1152.