PRIMT: A Pick-Revise Framework for Interactive Machine Translation

Shanbo Cheng, Shujian Huang, Huadong Chen, Xinyu Dai and **Jiajun Chen**
State Key Laboratory for Novel Software Technology
Nanjing University
Nanjing 210023, China
{chengsb, huangsj, chenhd, daixy, chenjj}@nlp.nju.edu.cn

Abstract

Interactive machine translation (IMT) is a method which uses human-computer interactions to improve the quality of MT. Traditional IMT methods employ a left-to-right order for the interactions, which is difficult to directly modify critical errors at the end of the sentence. In this paper, we propose an IMT framework in which the interaction is decomposed into two simple human actions: picking a critical translation error (Pick) and revising the translation (Revise). The picked phrase could be at any position of the sentence, which improves the efficiency of human computer interaction. We also propose automatic suggestion models for the two actions to further reduce the cost of human interaction. Experiment results demonstrate that by interactions through either one of the actions, the translation quality could be significantly improved. Greater gains could be achieved by iteratively performing both actions.

1 Introduction

To obtain high quality translations, human translators usually have to modify the results generated by a machine translation (MT) system (called post editing, PE). In many cases, PE needs a lot of modifications, which is time-consuming (Plitt and Masselot, 2010). To speed up the process, interactive machine translation (IMT) is proposed which instantly update the translation result after every human action (Langlais et al., 2000; Foster et al., 2002; Barrachina et al., 2009; Koehn, 2009; González-Rubio et al., 2013; Alabau et al., 2014). Because the translation quality could be improved after every update, IMT is expected to generate high quality translations with less human actions (Sanchis-Trilles et al., 2014).

Typical IMT systems usually use a left-to-right sentence completing framework pioneered by Langlais et al (2000), in which the users process the translation from the beginning of the sentence and interact with the system at the left-most error. By assuming the translation from the beginning to the modified part (called "prefix") to be correct, the system generates new translations after the given prefix (Koehn, 2009; Barrachina et al., 2009; Ortiz, 2011; Alabau et al., 2014).

Despite the success of this left-to-right framework, one potential weakness is that it is difficult to modify critical translation errors at the end of a sentence. Critical translation errors are those errors that has large impact on the translation of other words or phrases. When a translation ambiguity occurs at the end of a sentence while it causes translation errors at the beginning, modifying this critical errors first may bring great positive effects on previous parts of the translation, which may reduce human efforts in an IMT process. Modifying from left to right will delay the modification of the ambiguity point and lowers the interaction efficiency.

Critical errors are often caused by the inherent difficulty of translating source phrases. Mohit et al. (2007) proposed a classifier to identify the difficult-to-translate phrases (DTPs), which were extracted from syntactic trees. They demonstrated that asking human to translate these DTPs can bring a significant gain to the overall translation quality compared to translating other phrases. However, to our

Proceedings of NAACL-HLT 2016, pages 1240–1249,
San Diego, California, June 12-17, 2016.

Source	南亚 各国 外长 商讨 自由 贸易区 和 反 恐 问题
	(south asian)(countries)(foreign minister)(discuss) (free)(trade zone)(and)(anti)(terrorism)(issue)
Ref	south asian foreign ministers discuss free trade zone and anti-terrorism issues
Baseline	south asian foreign ministers to discuss the issue of free trade area and the
L2R	south asian foreign ministers discuss the issue of free trade area and the
PR	south asian foreign ministers **discuss free trade area and** anti-terrorism issues

Table 1: Examples of applying the Left-to-right (L2R) framework and the Pick-Revise framework (PR) in modifying a Chinese-English translation. Both the PR and left-to-right actions are performed only once. The first row shows the Chinese words and their translations. The following rows are the reference translations, the translation of the baseline system, the translation after a L2R interaction cycle and the translation after a PR interaction cycle, respectively. The dashed underline phrase is picked as the error to be modified by L2R. The underline phrase is picked as the error to be modified. The bold parts show the positive effects of revising the selected translation error on the translation of their contexts in a constrained decoding.

best knowledge, there is no practice in integrating these DTPs into an IMT framework.

In this paper, we propose a Pick-Revise IMT framework (PRIMT) to explicitly split the modification of a translation result into two very simple actions. Firstly, a wrongly-translated phrase is selected from the whole sentence (**Pick**); secondly, the correct translation is selected from the translation table (or manually added) to replace the original one (**Revise**). Our system then re-translates the sentence and searches for the best translation using previous modifications as constraints (Section 2). Furthermore, we propose two automatic suggestion models that could predict the wrongly-translated phrases and select the revised translation, respectively (Section 3). With the suggestion models, users only perform one of the actions (picking or revising) and let the suggestion models complete the other one. In this case, the interactions could be further simplified to be only one of the actions, which is as simple as one mouse click.

Experiment results show that by performing only one mouse click, the translation quality could be significantly improved (around +2 BLEU points in one PR cycle). Performing both two actions multiple times will bring greater gain in translation quality (+17 BLEU) with a relatively low Keystroke and Mouse-action Ratio (KSMR) (Barrachina et al., 2009) (3.3% KSMR).

2 The Pick-Revise IMT Framework

2.1 PRIMT System

We first explain the difference between Pick-Revise (PR) framework and left-to-right framework (Foster et al., 2002) with an example (in Table 1). For the given input source sentence, the MT system firstly generates a baseline translation. In the left-to-right framework human translator modifies the left-most error from "to discuss" to "discuss". But this modification may not bring any positive effects on the other part of the translation. So more interactions are needed to further improve the translation quality.

In our pick-revise framework, the human translator picks the phrase " 反 恐" which was considered the most critical translation error, and revise the translation from "the" to "anti-terrorism" according to phrase table. After a PR cycle, our constrained decoder re-translates the sentence. It not only generates the correct translation for the pick-revise pair (PRP), but also improves the translation around the PRP (bold parts).

Compared to left-to-right framework, our framework can modify the most critical error at first, which brings larger improvements on translation quality and improves the efficiency of human interactions.

Figure 1 shows an overview of our framework. For a source sentence $s_1 ... s_n$, our framework iteratively generates the translation using a constrained decoder. The constraints come from previous picking and revising processes. The picking and revising results can also be collected for model adaptation. The whole process continues until the translation is considered acceptable by the users. We explain the key components of our framework below.

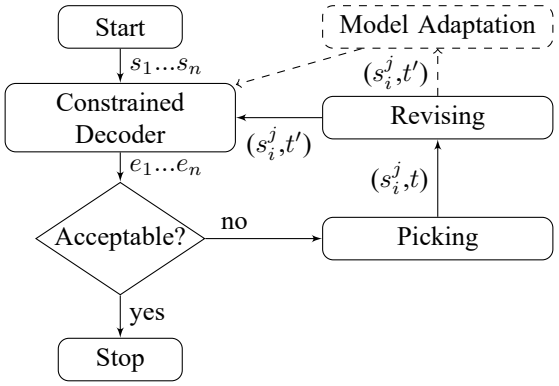

Figure 1: An overview of PRIMT framework.

2.2 Picking

In the picking step, the users pick the wrongly-translated phrase, (s_i^j, t)[1], to be revised. The picking process aims at finding critical errors in the translation, caused by errors in the translation table or inherent translation ambiguities. The more critical the error is, the larger translation quality improvement can be achieved by correcting the error (Mohit and Hwa, 2007). Critical errors might have a large influence to the translation of their context.

To make the picking step easier to be integrated into MT system, we limit the selection of translation errors to be those phrases in the previous PR-cycle output. If it's the first PR-cycle, then those errors come from phrases used to generate the baseline translation. For more convenient user interactions, in our PRIMT system, critical errors can be picked from both the source and target side by simply a mouse click on it. The correspondence/alignment between source and target phrases are visualized for easier human observation.

Green et al. (2014) demonstrated that performing post-editing, i.e. directly editing the translation errors, could get acceptable translations faster than performing left-to-right IMT. Such result also indicates that identifying critical translation errors is not a difficult task for human to perform.

2.3 Revising

In the revising step, the users revise the translation of s_i^j by selecting the correct translation t' from the translation table, or manually add one if there is no correct translation in the translation table. Whether to perform selection or adding depends on the quality of the translation table. When the translation system is trained with large enough parallel data, the quality of the translation table is usually high enough to offer the correct translation.

For a picked phrase, the translation options in the phrase table could be presented to the users as a list. The users just need to click on the correct translation to complete the revising step. The users could also type a new translation through a separated input area.

2.4 Decoder and Model Adaptation

A pick-revise pair (PRP), (s_i^j, t'), is obtained after a PR cycle for a source sentence. We use a constrained decoder to search for the best translation with the previous PRPs as constraints. The constrained search algorithm is similar to the algorithm in a typical phrase-based machine translation (Koehn et al., 2003). The only exception is that it makes an extra comparison between each translation option and previous PR pairs, which ignores all the phrases that overlap with the source side of a PRP. As a result, a lot of translation options are ignored, which makes the search space much smaller than standard decoding. In this way, we could guarantee that all the PRPs are correctly translated and the whole process can be carried out in real-time.

The system could collect all PRPs and adapt the models using methods described in Germann (2014) or Marie (2015). In our current implementation, we mainly focus on the picking and revising step and leave model adaptation as future work.

3 Automatic Suggestion Models

To further reduce the human actions, we propose to use automatic suggestion models for the picking and revising step, respectively. Such models can offer suggestions to users in both picking and revising steps. Because both picking and revising actions are performing selections from multiple candidates, we use classifier-based approaches to model these two steps. In the following subsections, we will introduce how we define the picking and revising tasks as classification tasks and how we choose features to model the tasks. Note that these automatic suggestion models could be interpreted as simplified confidence measurements.

[1] s_i^j is the phrase that covers the source words from index i to j, and translated into t.

3.1 The Picking Suggestion Model (PSM)

3.1.1 PSM Training

The picking process aims at selecting critical errors which has huge impact on the translation quality of their context. The goal of PSM is to automatically recognize those phrases that might be wrongly-translated, and suggest users to pick these phrases. In real world systems, the users can either accept or refuse the suggestion.

Within all the phrases of a source sentence, we need to separate the wrongly-translated phrases and correctly-translated phrases. Because translation errors often cause low translation quality, we use the translation quality gain after the revising action as a measurement. We treat those phrases that achieve translation quality improvement after revising as wrongly-translated phrases; those lead to translation quality deterioration as correctly-translated phrases.

We select phrases that lead to a BLEU improvement/deterioration greater than a threshold as positive/negative instances. In this paper, the threshold is set as 10% of the BLEU score of the baseline sentence.

3.1.2 PSM Features

Modeling the picking process needs two aspects of information. One of them is to determine whether the phrase is difficult-to-translate; the other is to determine whether the current translation option is correct. We use features from translation models (TMs), language models (LMs), lexical reordering models (LRMs), as well as counting and lexical features in Table 2. These features cover information of the source side, target side, translation ambiguity, and context, etc.

Category	Description
TM	TM scores of baseline translation
	Normalized TM scores of baseline translation
	TM entropy of all translation options
LM	LM score of baseline translation
	LM score of previous/next phrase translation
	LM score of each target word
	LM score of the bigram at the border of current and previous/next phrase
LRM	LRM scores of baseline translation
	LRM scores of previous/next phrase translation
Count	Source/target word count
	Number of translation options for current source phrase
POS	POS-tags of source words
	POS-tags of previous/next word of source phrase
Lexical	Source words
	Target words

Table 2: Features for the PSM.

3.2 The Revising Suggestion Model (RSM)

3.2.1 RSM Training

The revising process aims at selecting a correct translation for a given phrase under the given context. The goal of RSM is to predict the correct translation and suggest users to replace the wrong translation with the predicted one. The users can either accept it or use another translation.

Translation table has multiple translation options for one phrase. Within the translation option set of a source phrase, we need to separate the correct and wrong translation options. Instead of asking human translators to label these translations, we use two criteria to distinguish correct translation options from wrong translation options.

Firstly, the correct translation option should be a substring of the references, which ensures the correctness of the options itself. Secondly, the correct translation option should be consistent with pre-trained word alignment on the translated sentence pair[2]. This is to ensure that the translation option does not get credit for words that are not translations of the source side phrase. The remaining options are considered wrong translations.

With the above criteria, we select all correct translation options as positive instances for the revising step, and randomly sample the same number of wrong translation options to be negative instances. Specifically, translation options that are used by the baseline system are included as negative instances.

3.2.2 RSM Features

The features used for RSM are showed in Table 3. For translations of a given source phrase, there is no need to compare their source-side information because these translation options share the same source phrase and context. So these features mainly focus

[2]We trained word alignments with Giza++(Och and Ney, 2003)

on estimating the translation quality of a given translation option. As a result, features for RSM only including the scores for TM, LM and LRM, etc, which are simpler compared to PSM.

Category	Description
TM	TM scores of current translation option
LM	LM score of current translation option
	LM score of each target word
LRM	LRM scores of current translation option
count	Target word count
Lexical	Target words

Table 3: Features for the RSM

4 Experiments

4.1 Experiment Settings

4.1.1 Translation Settings

Through out the experiments, we use an in-house implementation of the phrase-based machine translation system (Koehn et al., 2003) and incorporate our PRIMT framework into the translation system. The parallel data for training the translation model includes 8.2 million sentences pairs from LDC2002E18, LDC2003E14, LDC2004E12, LDC2004T08, LDC2005T10, LDC2007T09. A 5-gram language model is trained with MKN smoothing (Chen and Goodman, 1999) on Xinhua portion of Gigaword which contains 14.6 million sentences. We use a combination of NIST02 and NIST03 to tune the MT system parameters and train the suggestion models. We test the system on NIST04 and NIST05 data. The translation results are evaluated with case insensitive 4-gram BLEU (Papineni et al., 2002). Our baseline phrase-based MT system has comparable performance with the open source toolkit Moses (Koehn et al., 2003).

4.1.2 Classification Settings

We use three classification models to model the automatic suggestion models: the maximum entropy model, the SVM model and the neural network model. We use a maximum entropy model (Zhang, 2004) with 30 iterations of L-BFGS. We use the LibSVM implementation (Chang and Lin, 2011) with RBF kernel and L2 regularization ($c = 128$, $\gamma = 0.5$). We use a feedforward neural network with the CNTK implementation (Agarwal et al., 2014). The neural network has one hidden layer of 80 nodes, with sigmoid function as the activation function.

We use one-hot representation for the source and target word features when using the maximum entropy and SVM model, and use pre-trained word embeddings (Mikolov et al., 2013) for the neural model.

4.2 Methodology

4.2.1 Simulated Human Interaction

Because real-world human interactions are expensive and time-consuming to obtain, we use simulated human interactions for picking and revising in the experiment.

Directly identifying critical errors in the translation is not an easy task without human annotation. Instead, we find critical errors by judging the influence of a given error to the translation of their context. We try picking every phrase in a baseline translation result and revising it using the simulated revising strategy (described below). The influence of the phrase is measured by the translation quality improvement after re-translation with the current phrase be revised. The phrase with the highest translation quality improvement is picked to be the simulated human picking result.

Given the phrase to be revised, the simulated revising action is straightforward. Among all the translation options that are considered correct (Sec. 3.2.1), we choose the longest one to be the simulated human revising result.

With the above simulated actions, one PR cycle takes exactly two mouse clicks and none key-stroke. For fair comparison, we use the same simulated revising action for the left-to-right framework. Each cycle of left-to-right framework also takes two mouse clicks. We also compare the post editing method which selects the most critical error and edits it to be the simulated revising translation. The key-stroke count for each editing is the number of characters of the correct phrase translation.

4.3 Translation Quality Improvement in Ideal Environment

Our first experiment is to test the PRIMT performance in an ideal environment. We conduct experiments on sentences for which the reference could be generated by our current MT system using forced decoding. Forced decoding forces the decoder to gen-

Data	NIST04(forced)		NIST05(forced)	
	BLEU	KSMR	BLEU	KSMR
Baseline	44.59	0	41.48	0
PR*1	**63.21 (+18.62)**	2.2	**55.10 (+13.62)**	2.2
PR*2	**70.82 (+26.23)**	4.3	**63.03 (+21.55)**	4.4
PR*3	**73.99 (+29.50)**	6.5	**68.56 (+27.08)**	6.7
PR*4	**75.48 (+30.89)**	8.6	**72.20 (+30.72)**	8.9
PR*5	**76.59 (+32.00)**	10.8	**73.90 (+32.42)**	11.1
PR*6	**78.07 (+33.48)**	12.9	**75.22 (+33.74)**	13.3
PR*7	**79.27 (+34.68)**	15.1	**75.57 (+34.09)**	15.5
PR*8	**79.54 (+34.93)**	17.2	**76.02 (+34.54)**	17.8
L2R*1	49.32 (+4.73)	2.2	46.34 (+4.86)	2.2
PE*1	49.77 (+5.18)	8.3	46.81 (+5.33)	8.2

Table 4: Experiments on sentences that can be forced-decoded for both NIST04 and NIST05 data, with 186 and 92 sentence counts, respectively. (PR*n denotes system that repeat picking and revising for n cycles; the PE system post edits the most critical error; the L2R system modifies the left most error).

erate translations exactly the same as the references. A reference translation could be generated by forced decoding means that it won't be necessary to input new words to generate a correct translation. Because we only simulate human revising actions as selecting the best translation option from phrase table (without adding new options), such a setting guarantees that the phrase table contains the correct translation for every phrase.

Table 4 shows that picking and revising the most critical error (PR*1) can bring +18 and +13 BLEU improvements in the two data sets, respectively. Revising the left-most error (L2R*1) only achieves an improvement around +5 BLEU. This result demonstrates that picking the critical error to be revised is critical in our PR framework. Compared to the left-to-right method, our framework has the advantage of correcting the critical errors in a high priority. By correcting such errors, the BLEU gain is much larger than left-to-right correction.

Post-editing the most critical error (PE*1) uses 8% KSMR, but only brings +5 BLEU improvement. Compared to post-editing, which just edits the critical error without affecting other parts of the translation, our PRIMT framework can re-decode for better translations with less human interactions.

In 8 PR-cycles (PR*8) (around 17% KSMR), the PRIMT achieves very high quality translation results with a BLEU score higher than 75 (around +35 BLEU to baseline). These results demonstrate the efficiency of PRIMT in multiple interactions.

Data	NIST04		NIST05	
	BLEU	KSMR	BLEU	KSMR
Baseline	31.83	0	30.64	0
PR*1	**42.88 (+11.05)**	1.1	**41.47 (+10.83))**	1.1
PR*2	**48.21 (+16.38)**	2.2	**45.76 (+15.12)**	2.2
PR*3	**50.12 (+18.29)**	3.3	**48.33 (+17.69)**	3.3
L2R*1	35.61 (+3.78)	1.1	33.85 (+3.21)	1.1
PE*1	34.74 (+2.91)	4.3	34.18 (+2.54)	4.8

Table 5: Experiments on both NIST04 and NIST05 data. (PR*n denotes system that repeat picking and revising for n cycles; the PE system post edits the most critical error; the L2R system corrects the left most error).

ASM	Classifier	NIST04	NIST05
PSM	MaxEnt	0.70/0.62/0.66	0.69/0.60/0.64
	SVM	0.71/0.68/0.69	0.69/0.66/0.67
	Feedforward	0.71/0.73/0.72	0.68/0.70/0.69
RSM	MaxEnt	0.71/0.58/0.63	0.70/0.57/0.63
	SVM	0.70/0.61/0.0.65	0.68/0.62/0.65
	Feedforward	0.66/0.67/0.66	0.65/0.65/0.65

Table 6: Classification performance of automatic suggestion models. The three values of each cell denotes the precision, recall and F-score, respectively, calculated on positive instances of corresponding classifier.

4.4 Translation Quality Improvement in General Environment

We also validate the improvements of translation quality in a general environment. We perform similar experiments on all NIST04 and NIST05 data. In some of the sentences, the translation table might not contain the correct translation for source phrase, due to the limitation of the training of our current MT system.

The results are listed in Table 5. Although the BLEU score in general environment are lower than those in ideal environment, the results show basically the same trends as in the previous experiment. The third row (PR*1) in Table 5 shows that picking and revising the most critical error can bring around +11 BLEU improvements in both data sets. The improvements in L2R*1 (+3.2) and PE*1 (+2.5) are much less. Three PR-cycles (around 3.3 KSMR) can achieve +17 BLEU improvements (PR*3). Compared to left-to-right and PE methods, our framework still has a significant advantage in the general environment.

4.5 Using Automatic Suggestion Models

We validate the effectiveness of our automatic suggestion models by both classification performance and translation performance.

Table 6 shows the classification performances of the PSM and the RSM, with different models. The precision and recall are calculated on positive instances in the test set, because only those instances that are predicted as positive will be used in the IMT system. Because it is harder to automatically identify the correct translation, we keep the translation unchanged when the RSM classifies all translation options to be negative.

The performance of the three classifiers are similar. Feedforward neural network has a moderate advantage. In general, the PSM could recognize the critical translation errors with an F-score around 0.67. The RSM achieves about 0.65 F-score for recognizing the correct translation. The F-scores are all in the range between 60 and 70, which is reasonable considering the difficulty of the tasks themselves.

We also evaluate the translation improvements when automatic suggestion models are used in the PR framework (Table 7). If the picking action performs a random pick of phrase (RandomPicking), there is barely no improvement in the translation quality, even with the simulated revising action. For comparison, using PSM could achieve a significant BLEU improvement of around 2 BLEU, on both test sets. It suggests that the BLEU gain does not come from the long reference translation match in the revising step. Picking critical errors is crucial in our framework.

Choosing the most critical error and performing a random revising action (RandomRevising) brings no improvement in BLEU either. Using our RSM could still improve the translation quality by 1.5 BLEU.

In general, using one of our PSM and RSM could still achieve significant improvement in translation quality. But the uses only need to perform one type of actions, which might be more suitable to be performed by a single human translator. However, the improvement is relatively small compared to fully simulated results, suggesting that human involvement is still critical for improve the translation quality. Better modeling or training with larger data may also improvement the performance of automatic suggestions.

		NIST04	NIST05
	Baseline	31.83	30.64
	RandomPicking	31.92 (+0.09)	30.69 (+0.05)
PSM	MaxEnt	33.89 (+2.06)	32.57 (+1.93)
	SVM	34.01 (+2.18)	32.66 (+2.02)
	FeedForward	34.23 (+2.40)	32.81 (+2.17)
	RandomRevising	31.90 (+0.07)	30.71 (+0.08)
RSM	MaxEnt	33.62 (+1.79)	32.38 (+1.74)
	SVM	33.73 (+1.90)	32.42 (+1.78)
	FeedForward	33.77 (+1.94)	32.44 (+1.80)

Table 7: Improvements of translation quality using random selection and automatic suggestion models.

5 Example Analysis

We further analyze the performance of our PRIMT system by examples. Table 8 shows the PRIMT procedure of improving translation quality for three different sentences.

In the first sentence, two PR cycles (4.7% KSMR) lead to a perfect translation. In the first PR cycle (PR*1), revising the translation of " 第六" from "the" to "the 6th" improves the neighboring translation. The translation of " 证实" change from "confirms" to "confirm", which is a positive effect. In PR*2, revising the translation of " 病例" from "cases" to "case" also changes the neighborhood translation (the translation of " 禽流感 死亡 病例" changes to "death case from the bird flu"). After two PR cycles, the reference translation is obtained.

In our current settings, the reference translation could not always be obtained. The maximum achievable BLEU is around 60-70 in general environment. The next two examples shows some possible explanations.

In the second sentence in Table 8, " 需要 一定" is picked in the first PR cycle. Revising the translation from "a" to "need a certain" makes the translation of " 通常" changing from "is" to "usually". In the next PR cycle, revising the translation of " 过程" from "process" to "course" makes the neighboring translation changing from "," to ", and". Meanwhile, the position of "course" moves to the right place (in front of ","). In the last PR cycle, the translation of " 很难" is revised from "it" to "it cannot be". After three PR cycles, the translation quality improves significantly. However, the translation is still different from the

Source	世卫 组织 证实 越南 第六[1] 个 禽流感 死亡 病例[2]
	(world health) (organization) (confirm) (vietnam) (6th) () (bird flu) (death) (case)
Ref	the world health organization confirms the 6th death case from the bird flu in vietnam
Baseline	the world health organization confirmed the bird flu death cases in vietnam
PR*1	the world health organization **confirms** the 6th[1] bird flu death cases in vietnam
PR*2	the world health organization confirms the 6th death case[2] **from the bird flu** in vietnam
Source	民族 和解 通常 需要 一定[1] 的 过程[2] ， 很 难[3] 一蹴而就[4] 。
	(national) (reconciliation) (usually) (need) (certain) () (course) (,) (cannot) (accomplish in one action) (.)
Ref	national reconciliation usually need a certain course , and it cannot be accomplished in one action .
Baseline	national reconciliation is a very difficult process takes .
PR*1	national reconciliation process **usually** need a certain[1] takes very difficult .
PR*2	national reconciliation **usually need a certain** course[2] **, and** it accomplished .
PR*3	national reconciliation usually need a certain course , and it cannnot be[3] accomplished .
Human	national reconciliation usually need a certain course , and it cannnot be accomplished in one action[4] .
Source	然而， 以色列的[2] 回答 无法[1] 充分扫除 美国 的 疑问 。
	(however) (israel's) (reply) (fail) (full clear) (the us) () (doubt) (.)
Ref	however , israel 's reply failed to fully clear the us doubts .
Baseline	however , the israeli response to the full removal of united states .
PR*1	however , the israeli response failed to[1] **fully clear** doubts . the us
PR*2	however , israel 's[2] **reply** failed to fully clear doubts . the us

Table 8: Examples of applying PR actions multiple times in the Chinese-English translation. The superscript i of the underline phrases, P_i in source sentence denotes the underline phrase is picked in the i-th PR cycle as the critical error. The original translation of P_i is the underline phrase without superscript in PR*($i-1$) (Baseline is PR*0). The correct translation is the underline parts with superscript in PR*i. The bold parts shows the positive effects on other parts near the PRPs.

reference. This is because " 一蹴而就" should be translated into "accomplished in one action" instead of "accomplished". But there is no suitable translation options for it in the current phrase table. So the system cannot generate a perfect translation. The problems will be less significant when real-world human translators are involved. Human translator inputs the correct translation "accomplished in one action", the system will generate the reference translation after constrained decoding (Human).

In the last sentence in Table 8, " 无法" is picked as the critical error. Revising the translation from "to the" to "failed to", leads to an improvement on neighboring phrase (the translation of " 充分扫除" to "fully clear"). In the second PR cycle, " 以色列的" is picked. Revising the translation from "the israeli" to "israel 's", makes the translation of " 回答" change from "response" to "reply", which is also a positive effect. However, after two PR cycles, all phrase translations are correct, but the translation is still different from the reference. This is because the language model and lexical reordering model prefer the wrong phrase ordering, which put "the us" at the end of the whole sentence. This problem raises from the MT system itself, which may not be solved directly in our current framework.

If more interactions are allowed, for example, performing reordering operations, the above problems could be solved. But the interactions become more complex, and may not be acceptable to human translators. Other solutions includes using better statistical models such as neural language models (Bengio et al., 2003). This is an interesting issue we will look into.

6 Conclusion

We introduced a pick-revise IMT framework, PRIMT, where the users could pick critical translation errors anywhere in the sentence and revise the translation. By correcting the critical error instead of the left most one, our framework could improve the translation quality in a quicker and more efficient way. By using automatic suggestion models, we could reduce human interaction to a single type, either picking or revising. It is also possible to let different human translators to perform different action-

s. In this case every translator will focus on a single action, which might be easier to train and may have higher efficiency.

On the other hand, the performance of current framework is still related to the underlying MT system. Further improvement could be achieved by supporting other type of interactions, such as reordering operations, or building the system with stronger statistical models. We will also conduct real-world experiments to see how this new IMT framework works when human translators are actually involved.

7 Acknowledgement

The authors would like to thank the anonymous reviewers for their valuable comments. This work is supported by the National Natural Science Foundation of China (No. 61300158, 61472183), the Jiangsu Provincial Research Foundation for Basic Research (No. BK20130580). This research is partially supported by the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University. Shujian Huang is the corresponding author.

References

Amit Agarwal, Eldar Akchurin, Chris Basoglu, Guoguo Chen, Scott Cyphers, Jasha Droppo, Adam Eversole, Brian Guenter, Mark Hillebrand, Xuedong Huang, Zhiheng Huang, Vladimir Ivanov, Alexey Kamenev, Philipp Kranen, Oleksii Kuchaiev, Wolfgang Manousek, Avner May, Bhaskar Mitra, Olivier Nano, Gaizka Navarro, Alexey Orlov, Marko Padmilac, Hari Parthasarathi, Baolin Peng, Alexey Reznichenko, Frank Seide, Michael L. Seltzer, Malcolm Slaney, Andreas Stolcke, Huaming Wang, Kaisheng Yao, Dong Yu, Yu Zhang, and Geoffrey Zweig. 2014. An introduction to computational networks and the computational network toolkit. Technical Report MSR-TR-2014-112, August.

Vicent Alabau, Christian Buck, Michael Carl, Francisco Casacuberta, M Garcıa-Martınez, Ulrich Germann, Jesús González-Rubio, Robin Hill, Philipp Koehn, LA Leiva, et al. 2014. Casmacat: A computer-assisted translation workbench. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 25--28.

Sergio Barrachina, Oliver Bender, Francisco Casacuberta, Jorge Civera, Elsa Cubel, Shahram Khadivi, Antonio Lagarda, Hermann Ney, Jesús Tomás, Enrique Vidal, et al. 2009. Statistical approaches to computer-assisted translation. *Computational Linguistics*, 35(1):3--28.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The Journal of Machine Learning Research*, 3:1137--1155.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Stanley F Chen and Joshua Goodman. 1999. An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, 13(4):359--393.

George Foster, Philippe Langlais, and Guy Lapalme. 2002. User-friendly text prediction for translators. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 148--155. Association for Computational Linguistics.

Ulrich Germann. 2014. Dynamic phrase tables for machine translation in an interactive post-editing scenario. In *AMTA 2014 Workshop on Interactive and Adaptive Machine Translation, Vancouver, BC, Canada*, pages 20--31.

Jesús González-Rubio, Daniel Ortiz-Martínez, José-Miguel Benedí, and Francisco Casacuberta. 2013. Interactive machine translation using hierarchical translation models. In *Conference on Empirical Methods in Natural Language Processing*, pages 244--254.

Spence Green, Sida I Wang, Jason Chuang, Jeffrey Heer, Sebastian Schuster, and Christopher D Manning. 2014. Human effort and machine learnability in computer aided translation. In *Conference on Empirical Methods in Natural Language Processing*, pages 1225--1236.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48--54. Association for Computational Linguistics.

Philipp Koehn. 2009. A web-based interactive computer aided translation tool. In *Proceedings of the ACL-IJCNLP 2009 Software Demonstrations*, pages 17--20. Association for Computational Linguistics.

Philippe Langlais, George Foster, and Guy Lapalme. 2000. Transtype: a computer-aided translation typing system. In *Proceedings of the 2000 NAACL-ANLP Workshop on Embedded machine translation systems-Volume 5*, pages 46--51. Association for Computational Linguistics.

Benjamin Marie, Lingua et Machina, France Le Chesnay, and Aurélien Max. 2015. Touch-based pre-post-editing of machine translation output. In *Conference*

on Empirical Methods in Natural Language Processing.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111--3119. Neural Information Processing Systems.

Behrang Mohit and Rebecca Hwa. 2007. Localization of difficult-to-translate phrases. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 248--255. Association for Computational Linguistics.

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

Daniel Ortiz. 2011. *Advances in fully-automatic and interactive phrase-based statistical machine translation*. Ph.D. thesis, Universitat Politècnica de València.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311--318. Association for Computational Linguistics.

Mirko Plitt and François Masselot. 2010. A productivity test of statistical machine translation post-editing in a typical localisation context. *The Prague Bulletin of Mathematical Linguistics*, 93:7--16.

Germán Sanchis-Trilles, Vicent Alabau, Christian Buck, Michael Carl, Francisco Casacuberta, Mercedes García-Martínez, Ulrich Germann, Jesús González-Rubio, Robin L Hill, Philipp Koehn, et al. 2014. Interactive translation prediction versus conventional post-editing in practice: a study with the casmacat workbench. *Machine Translation*, 28(3-4):217--235.

Le Zhang. 2004. Maximum entropy modeling toolkit for python and c++.