Neural Pre-Translation for Hybrid Machine Translation

Jinhua Du jinhua.du@adaptcentre.ie
Andy Way andy.way@adaptcentre.ie
ADAPT, School of Computing, Dublin City University, Ireland

Abstract

Hybrid machine translation (HMT) takes advantage of different types of machine translation (MT) systems to improve translation performance. Neural machine translation (NMT) can produce more fluent translations while phrase-based statistical machine translation (PB-SMT) can produce adequate results primarily due to the contribution of the translation model. In this paper, we propose a cascaded hybrid framework to combine NMT and PB-SMT to improve translation quality. Specifically, we first use the trained NMT system to pre-translate the training data, and then employ the pre-translated training data to build an SMT system and tune parameters using the pre-translated development set. Finally, the SMT system is utilised as a post-processing step to re-decode the pre-translated test set and produce the final result. Experiments conducted on Japanese→English and Chinese→English show that the proposed cascaded hybrid framework can significantly improve performance by 2.38 BLEU points and 4.22 BLEU points, respectively, compared to the baseline NMT system.

1 Introduction

In recent years, NMT has made impressive progress (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2015). The state-of-the-art NMT model employs an encoder–decoder architecture with an attention mechanism, in which the encoder summarizes the source sentence into a vector representation, the decoder produces the target string word by word from vector representations, and the attention mechanism learns the soft alignment of a target word against source words (Bahdanau et al., 2015). NMT systems have outperformed the state-of-the-art SMT model on various language pairs in terms of translation quality (Luong et al., 2015a; Bentivogli et al., 2016; Junczys-Dowmunt et al., 2016; Wu et al., 2016; Toral and Sánchez-Cartagena, 2017). However, due to some deficiencies of NMT systems such as the limited vocabulary size, and meaningless translations, much research work has involved combining NMT and SMT to improve translation performance (Cho et al., 2014; He et al., 2016; Niehues et al., 2016; Wang et al., 2017).

HMT is a strategy that combines different types of translation systems and fully takes advantage of the strengths of each system to improve translation performance. A typical example of HMT involves a rule-based system's predictable and consistent translations with an SMT system used for post-processing to further improve translation quality (Groves and Way, 2005; Paul et al., 2005; Groves and Way, 2006; Chen et al., 2007; Sánchez-Martínez et al., 2007; Enache et al., 2012; Li et al., 2015).[1] NMT is a new MT paradigm which can produce highly

[1] A huge amount of work has been done on this topic in the past decades, so we only list a representative sample here.

fluent translations. However, NMT sometimes generates translations that have a totally different meaning compared to the source sentences, which testifies to its strong language modeling but weak translation modeling capabilities. By contrast, PB-SMT is good at reflecting the adequacy of source sentences by means of the 'hard word alignment'. Intuitively, if we take the fluent but wrong translations as another language and perform word alignment, then we can perhaps restore the original meaning of source sentences to some extent using SMT. Moreover, for any source-side out-of-vocabulary (OOV) words in NMT, if we can keep them in the translations, then we can fully utilise the advantage of SMT to translate them.

Some work has been done on combining the adequacy of SMT and fluency of NMT. To the best of our knowledge, the most similar work is the pre-translation framework proposed in Niehues et al. (2016). In their framework, the SMT system is first used to pre-translate the input and then an NMT system generates the final hypothesis using the pre-translation as input. However, in their experiments, the "SMT⇒NMT" framework without integrating source information did not beat the pure NMT system and was not able to combine the strengths of both systems.

In this paper, we propose an "NMT⇒SMT" hybrid strategy to utilise SMT and NMT by considering (i) that NMT systems significantly outperform SMT systems, so using a higher-quality system as a post-processing step may indeed improve the performance of a lower-quality MT system, but might be difficult to outperform the higher-quality system; (ii) that NMT is more sensitive to noisy data compared to SMT, so using pre-translated data to train NMT will cause translation performance to deteriorate. Accordingly, the NMT system trained on the pre-translated data will not be able to correct errors from the SMT system. Experiments conducted on Japanese→English and Chinese→English demonstrate that our proposed "NMT⇒SMT" hybrid strategy can alleviate the above problems and further improve translation quality compared to pure NMT systems. The main contributions of this work include:

- We re-implement the "SMT⇒NMT" strategy on two different language pairs and four directions, namely Japanese↔English and Chinese↔English. Results show that this framework indeed cannot outperform the baseline NMT system.

- We propose an "NMT⇒SMT" hybrid framework that can better combine SMT and NMT by using their different strengths.

- We examine the effectiveness of the proposed framework on different NMT systems, namely the single NMT, factored NMT and ensemble NMT systems.

The rest of the paper is organised as follows. In Section 2, related work to the proposed neural hybrid MT framework is introduced. Section 3 describes the attentional encoder–decoder framework for NMT, and Section 4 introduces factored NMT and our proposed input features for NMT. In Section 5, we detail our proposed neural hybrid MT framework. In Section 6, we report the results of two sets of experiments on Chinese–English and Japanese–English tasks. Then a qualitative analysis is carried out, and some examples for comparing different systems are also illustrated in this section. Section 7 concludes and gives avenues for future work.

2 Related Work

The combination of NMT and SMT can be roughly categorised into three categories:

- NMT in post-processing: in this scenario, translations from SMT can be post-processed using NMT. For example, using NMT or neural networks to re-rank the outputs from SMT (Zhao et al., 2014; Lee et al., 2015; Neubig et al., 2015; Ding et al., 2016; Farajian et al., 2016), or using pre-translated results from SMT to build an NMT system (Niehues et al., 2016).

- Integrating SMT into NMT: in this scenario, SMT is used to guide translation in NMT, e.g. incorporating the translation model or language model into the decoding process of NMT (He et al., 2016; Wang et al., 2017).

- Integrating NMT into SMT: in this category, it is essentially integrating neural network-based features into SMT, such as the neural reordering model, neural language model, neural semantic model etc., e.g. Cho et al. (2014); Li et al. (2014); Passban et al. (2015).

In terms of the second category, He et al. (2016) incorporate SMT features, such as a translation model and an *n*-gram language model, with the NMT model under the log-linear framework. Their experiments show that the proposed method significantly improves translation quality of the baseline NMT system on Chinese→English translation tasks.

Wang et al. (2017) propose to incorporate an SMT model into the NMT framework in which at each decoding step, SMT offers additional recommendations of generated words based on the decoding information from NMT, and then an auxiliary classifier is employed to score the SMT recommendations and a gating function is used to combine the SMT recommendations with NMT generations, both of which are jointly trained within the NMT architecture in an end-to-end manner. Experimental results on Chinese-to-English translation show that the proposed approach achieves significant and consistent improvements over state-of-the-art NMT and SMT systems.

The proposed hybrid framework in this paper can be defined as a novel fourth category where we use SMT to post-process translations from NMT, which is completely different from the "SMT⇒NMT" framework in Niehues et al. (2016). In their framework, the SMT system is used to pre-translate the input and then an NMT system generates the final hypothesis using the pre-translation. In their experiments, the basic pre-translation system did not beat the NMT system either on natural order or pre-reordered data. By concatenating the source-side sentences with pre-translations as input to NMT, the final translation performance outperformed the baseline NMT system. From their results, we can see that the framework still needs the source information, and it is difficult to tell whether the improvements are mainly contributed by the pre-translation or source information.

3 Neural Machine Translation

The basic principle of an NMT system is that it can map a source-side sentence $\mathbf{x} = (x_1, \ldots, x_m)$ to a target sentence $\mathbf{y} = (y_1, \ldots, y_n)$ in a continuous vector space, where all sentences are assumed to terminate with a special "end-of-sentence" token $< eos >$. Conceptually, an NMT system employs neural networks to solve the conditional distributions in (1):

$$p(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{n} p(y_i|y_{<i}, x_{\leq m}) \tag{1}$$

We utilise the NMT architecture in Bahdanau et al. (2015), which is implemented as an attentional encoder-decoder network with recurrent neural networks (RNN).

In this framework, the encoder is a bidirectional neural network (Sutskever et al., 2014) with gated recurrent units (Cho et al., 2014) where a source-side sequence $\mathbf{x}$ is converted to a one-hot vector and fed in as the input, and then a forward sequence of hidden states $(\overrightarrow{h}_1, \ldots, \overrightarrow{h}_m)$ and a backward sequence of hidden states $(\overleftarrow{h}_1, \ldots, \overleftarrow{h}_m)$ are calculated and concatenated to form the annotation vector h_j. The decoder is also an RNN that predicts a target sequence $\mathbf{y}$ word by word where each word y_i is generated conditioned on the decoder hidden state s_i, the previous target word y_{i-1}, and the source-side context vector c_i, as in (2):

$$p(y_i|y_{<i}, \mathbf{x}) = g(y_{i-1}, s_i, c_i) \tag{2}$$

where g is the activation function that outputs the probability of y_i, and c_i is calculated as a weighted sum of the annotations h_j. The weight α_{ij} is computed as in (3):

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{m} exp(e_{ik})} \tag{3}$$

where

$$e_{ij} = a(s_{i-1}, h_j)$$

is an alignment model which models the probability that the inputs around position j are aligned to the output at position i. The alignment model is a single-layer feedforward neural network that is learned jointly through backpropagation.

4 Factored NMT Using Linguistic Features

Factored NMT, introduced in Sennrich and Haddow (2016), represents the encoder input as a combination of features as in (4):

$$\overrightarrow{h}_j = g(\overrightarrow{W}(\mathop{\Big\|}_{k=1}^{|F|} E_k x_{jk}) + \overrightarrow{U}\overrightarrow{h}_{j-1}) \tag{4}$$

where $\|$ is the vector concatenation, $E_k \in \mathbb{R}^{m_k \times K_k}$ are the feature-embedding matrices, with $\sum_{k=1}^{|F|} m_k = m$, K_k is the vocabulary size of the k_{th} feature, and $|F|$ is the number of features in the feature set F (Sennrich and Haddow, 2016).

In factored NMT, the features can be any form of knowledge which might be useful to NMT systems, such as POS tags, lemmas, morphological features and dependency labels as used in Sennrich and Haddow (2016). In our work, besides POS tags, we also use a new feature – word class (WoC) – in the NMT system. We define “POS+WoC” as pre-reordering features because they are used for pre-reordering source-side sentences in SMT (Neubig et al., 2012; Nakagawa, 2015).

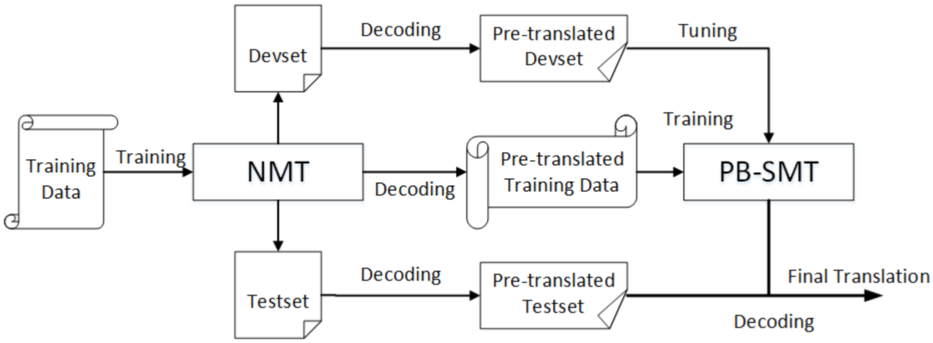

Figure 1: The Cascaded framework for neural HMT

5 Cascaded Hybrid Machine Translation

NMT can produce more fluent translations than SMT. However, NMT often produces some meaningless translations, i.e. the translation is totally different from the original meaning of the source sentence (Toral and Sánchez-Cartagena, 2017), or repeatedly translates some source words while mistakenly ignoring other words (Tu et al., 2017). We infer that this problem is due to the lack of an explicit translation model in NMT and the whole framework of NMT is regarded as a language model.

Although the soft-attention mechanism is helpful to guide the prediction of target words using the source information, and a reconstructor in NMT can manage to reconstruct the input source sentence from the hidden layer of the output target sentence to avoid the duplicate translation of source words (Tu et al., 2017), it still cannot explicitly and fully use the source information as in SMT. Thus, if we regard the translation from NMT as another 'language', and use SMT to perform word alignment and build a translation model, we might alleviate the meaningless and duplicate translations to some extent.[2]

Therefore, we propose an "NMT⇒SMT" framework to combine NMT and SMT as a multi-engine hybrid MT system as illustrated in Figure 1. In this pipeline, the first step is to train an NMT system using the initial training data, and then translate the training data, development set (devset) and test set (testset) into pre-translations; the second step is to use the pre-translated training data to build a target–target SMT system and tune the parameters using the pre-translated devset; the last step is to use the tuned SMT system to decode the pre-translated test set and produce the final output.

During pre-translation using NMT to translate the training data, devset and testset, we allow NMT to generate the 'UNK' token if an OOV occurs in the source sentence. Then, we propose a very simple but effective method to replace the "UNK" token in the translation by the corresponding source word. The method is shown in Algorithm 1.

Algorithm 1 Replacing UNK by source words

```
Require: A source sentence $f_1^l$, the translation $e_1^m$ with UNK tokens, and the limited source-side vocabulary $V$ for NMT.
  source_position = 1
  for $i = 1$ to $m$ do
    if $e_i$ == UNK then
      for $j$ =source_position to $l$ do
        if $f_j$ not in $V$ then
          $e_i$ = $f_j$
          source_position = $j + 1$
          break
        end if
      end for
    end if
  end for
```

The mechanism of Algorithm 1 is different from that in Jean et al. (2015) where they use the soft word-alignment information from the NMT system to guide the substitution process. However, generating the word-alignment information from NMT is quite time-consuming, especially for the translation of the training data. Therefore, our algorithm simply traverses the

[2] This is an open question. Intuitively, this depends on what word alignments are learned and what phrase pairs are extracted.

translation and its corresponding source sentence. Specifically, when we encounter the 'UNK' token, we will look up the source words in order in the NMT source-side vocabulary. If the source word does not exist in the vocabulary, then we replace the 'UNK' with this source word. We repeat this process to the end of the translation sentence. There might exist the wrong replacement due to different word order between the source sentence and the target sentence.[3] However, we believe that the SMT system will use its local reordering capability to reorder the OOVs to some extent and translate them.

Finally, different from using a back-off dictionary to post-process these unknown words in Luong et al. (2015b), we employ an SMT system to translate by considering more context.

6 Experiments

As Japanese and Chinese languages differ drastically from English in terms of word order and grammatical structure, we select Japanese–English and Chinese–English translations[4] to verify the proposed framework.

We also re-implement the "SMT⇒NMT" pipeline proposed in Niehues et al. (2016) as a comparison with our proposed framework. Therefore, two sets of experiments are set up as follows:

- "SMT⇒NMT": four translation directions (JP↔EN and ZH↔EN) are evaluated on natural-order and pre-reordered data. We employ the top-down BTG-based pre-reordering method to reorder source-side sentences (Nakagawa, 2015).
- "NMT⇒SMT": we test our proposed framework by integrating different types of NMT systems on JP→EN and ZH→EN tasks.

In the following sections, we report our experimental setup and results in terms of these two experiments.

6.1 Experimental Settings

For JP–EN translation tasks, the training data is the first part (train-1) of the JP–EN Scientific Paper Abstract Corpus (ASPEC-JE) that contains 1M sentence pairs, the development/validation set contains 1,790 sentence pairs, and the test set contains 1,812 sentence pairs (Nakazawa et al., 2016). There is only one reference for each source-side sentence in the validation and test sets.

For ZH–EN tasks, we use 1.4M sentence pairs extracted from LDC ZH–EN corpora as the training data, and NIST 2004 current set as the development/validation set that contains 1,597 sentences, and NIST 2005 current set as the test set that contains 1,082 sentences. There are four references for each Chinese sentence and there is only one reference for each English sentence in the validation and test sets. For the EN→ZH direction, we use the first reference out of four references for Chinese as the input (English).

For factored NMT, we use POS tags and word class as input features, which are obtained as follows:

- POS tag: the Japanese data are segmented and tagged using KyTea (Neubig et al., 2011), and the Chinese data are segmented and tagged using the ICTCLAS toolkit (Zhang et al., 2003).
- Word Class (WoC): the word classes of the training data are obtained using "mkcls" by setting the number of classes to 50. For an OOV word in the validation and test sets, we randomly allocate a class between (1, 50) to it.

[3] Noting that the number of OOVs in the source sentence is not always precisely the same as the number of UNKs in the translation of NMT. In our method, we take the minimum of these two numbers to replace the OOVs.

[4] In the rest of the paper, we use JP, ZH and EN to denote Japanese, Chinese and English, respectively.

Chinese and Japanese are not suitable for using the Byte Pair Encoding (BPE) method (Sennrich et al., 2016) to encode words as subword units, so we keep the words as translation units. We use Moses (Koehn et al., 2007) with default settings as the standard PB-SMT system, and use KenLM (Heafield et al., 2013) to train a 5-gram language model. We use Nematus (Sennrich et al., 2017) as the NMT system, and set minibatches of size 80, a maximum sentence length of 60, word embeddings of size 600, and hidden layers of size 1024. The vocabulary size for input and output is set to 45K. The models are trained with the Adadelta optimizer (Zeiler, 2012), reshuffling the training corpus between epochs. We validate the model every 5,000 minibatches via BLEU (Papineni et al., 2002) scores on the validation set and save the model every 30,000 iterations.

As in Sennrich and Haddow (2016), for factored NMT systems, in order to ensure that performance improvements are not simply due to an increase in the number of model parameters, we keep the total size of the embedding layer fixed to 600. Tables 1 and 2 show the vocabulary size and embedding size for pre-reordering features and the word as the input for the JP→EN and ZH→EN systems, respectively.

	Vocab. Size		Embedding Size	
Feature	Corpus	Model	All	Single
POS	21	21	10	10
WoC	51	51	10	10
Word	161,390	45K	580	590

Table 1: Vocabulary size, and size of embedding layer of pre-reordering features and words for JP→EN

	Vocab. Size		Embedding Size	
Feature	ZH	Model	All	Single
POS	36	36	10	10
WoC	51	51	10	10
Word	185,029	45K	580	590

Table 2: Vocabulary size, and size of embedding layer of pre-reordering features and words for ZH→EN.

In Tables 1 and 2, the columns from left to right under "Vocab. Size" indicate the vocabulary size of each feature. For example, "21" for "Corpus" indicates that there are a total of 21 POS tags in our Japanese corpus, and "21" for "Model" indicates that the vocabulary size of POS tags configured in the NMT model is 21. The column "All" indicates the embedding size of a feature when it combines with all other features, and "Single" indicates the embedding size of a feature only combining with "Word". In all NMT systems, the total embedding size is fixed to 600. Therefore, "590" indicates that for each single feature, the word embedding size for "Word" is obtained by [$600 - embedding\_size(feature) = 600 - 10 = 590$].

The NMT system with the best BLEU score is selected as our baseline, and in terms of the ensemble NMT system, we use the last 5 models. The beam size for all NMT systems is set to 12.

We only employ the pre-translated training data and devset from the baseline NMT system to train and tune the SMT engine. Then the tuned SMT system is employed to re-decode the pre-translated test set using the baseline NMT, factored NMT and ensemble NMT systems, respectively.

	JP→EN				EN→JP			
	Non-reordered		Pre-reordered		Non-reordered		Pre-reordered	
SYS	Validation	Test	Validation	Test	Validation	Test	Validation	Test
SMT	18.25	17.64	**21.79***	**21.71***	27.03	26.32	**33.67***	**33.75***
NMT	**24.16***	**24.55***	20.42	21.43	**35.25***	**35.23***	32.75	32.98
SMT⇒NMT	18.01	17.83	20.39	20.91	27.64	27.57	33.23	33.43

Table 3: Results on JP–EN SMT⇒NMT experiments. "*" indicates translation performance is significantly better.

	ZH→EN				EN→ZH			
	Non-reordered		Pre-reordered		Non-reordered		Pre-reordered	
SYS	Validation	Test	Validation	Test	Validation	Test	Validation	Test
SMT	33.13	29.24	**34.63***	**30.59***	14.50	12.77	**16.12***	**13.77***
NMT	**35.49***	**31.76***	33.95	30.23	**15.97***	**15.62***	14.14	13.53
SMT⇒NMT	32.87	28.86	33.84	29.69	14.51	12.94	15.45	13.36

Table 4: Results on ZH–EN SMT⇒NMT experiments

All results are reported by case-insensitive BLEU scores and statistical significance is calculated via a bootstrap resampling significance test (Koehn, 2004).

6.2 Results and Analysis on SMT⇒NMT

Tables 3 and 4 show the results for JP↔EN and ZH↔EN with and without pre-reordered data, respectively. The baseline system is a standard PB-SMT system trained on non-reordered and pre-reordered data, respectively. "NMT" indicates the baseline NMT system as described in Section 6.1.

From Table 3, we can see that:

- Non-reordered task: all "SMT⇒NMT" systems on JP→EN and EN→JP are significantly worse than the baseline NMT systems. Except for the validation set of JP→EN, all other "SMT⇒NMT" systems on JP→EN and EN→JP outperform the baseline SMT systems.

- Pre-reordered task: the "SMT⇒NMT" system is worse than both the pre-reordered NMT system and pre-reordered SMT system on JP→EN, while it is better than the pre-reordered NMT system on EN→JP.

For ZH↔EN tasks, the "SMT⇒NMT" system performs worse relative to JP↔EN tasks, i.e. almost all "SMT⇒NMT" systems did not beat the NMT and SMT systems.

The observations from these experiments show that:

- if the translation quality of NMT is better than that of SMT, then using NMT as a post-processing module without integrating source-side information to re-decode translations from SMT cannot further improve translation performance.

- the pre-reordering in the source-side sentences is indeed helpful to SMT, while it hurts the performance of NMT.

- we need a better pipeline to combine NMT and SMT without using the source-side information.

6.3 Results and Analysis on NMT⇒SMT

Results on the proposed NMT⇒SMT model are shown in Table 5, where "NMT⇒SMT-B" indicates that the "NMT⇒SMT" pipeline re-decodes the translations of the baseline NMT system, "NMT⇒SMT-F" indicates that the "NMT⇒SMT" pipeline re-decodes the translations of the factored NMT system, and "NMT⇒SMT-E" indicates that the "NMT⇒SMT" system re-decodes the translations of the ensemble NMT system.[5]

	JP→EN		ZH→EN	
SYS	Validation	Test	Validation	Test
SMT	18.25	17.64	33.13	29.24
NMT	24.16	24.55	35.49	31.76
NMT⇒SMT-B	25.33*	25.66*	36.58*	32.38*
factored NMT	25.08	25.17	37.42	33.15
NMT⇒SMT-F	25.94*	26.08*	37.69*	33.39*
ensemble NMT	26.24	26.37	39.10	35.69
NMT⇒SMT-E	26.80*	26.93*	39.53*	35.98*

Table 5: Results of SMT⇒NMT experiments on JP→EN and ZH→EN. "*" indicates translation performance is significantly better.

We observe that:

- JP→EN: the NMT⇒SMT-B improves translation performance by 1.17 BLEU points and 1.11 BLEU points on validation and test sets, respectively, compared to the baseline NMT system. The NMT⇒SMT-F improves by 0.86 BLEU points and 0.91 BLEU points on the validation and test sets, respectively, compared to the factored NMT system. The NMT⇒SMT-E improves by 0.56 BLEU points and 0.56 BLEU points on the validation and test sets, respectively, compared to the ensemble NMT system, and improves by **2.64** BLEU points and **2.38** BLEU points on the validation and test sets, respectively, compared to the baseline NMT system. All improvements are significantly better.

- ZH→EN: the NMT⇒SMT-B improves translation performance by 1.09 BLEU points and 0.62 BLEU points on validation and test sets, respectively, compared to the baseline NMT system. The NMT⇒SMT-F improves by 0.27 BLEU points and 0.24 BLEU points on the validation and test sets, respectively, compared to the factored NMT system. The NMT⇒SMT-E improves by 0.43 BLEU points and 0.29 BLEU points on the validation and test sets, respectively, compared to the ensemble NMT system, and improves by **4.04** BLEU points and **4.22** BLEU points on the validation and test sets, respectively, compared to the baseline NMT system. All improvements are significantly better.

The results show that:

- Our proposed neural hybrid MT pipeline is more effective and feasible than the SMT⇒NMT pipeline. In Niehues et al. (2016), the SMT⇒NMT pipeline only works when integrating the source information into NMT. However, it increases the computational complexity by concatenating the pre-translated and source sentences as input to NMT.

[5] In current experiments, we only ensemble the baseline NMT systems. In future, we also plan to ensemble the factored NMT systems to verify the HMT performance.

- Our proposed NMT⇒SMT framework only uses source-side information once, i.e. at the stage of NMT training, while at the stage of post-processing, we only use the pre-translations without the source information (except OOVs), which keeps the framework simpler than the SMT⇒NMT framework.
- For different types of NMT systems, the proposed pipeline can significantly further improve translation performance, and the pre-translated SMT system is only trained using translations from the baseline NMT system. We would expect further improvements if we use the translations from the factored NMT or ensemble NMT models to train the SMT engine.

From the analysis on the results, we found that:

- OOVs rate in the test set is significantly decreased in the proposed framework, i.e. the post-processing SMT system can translate some of the OOVs appearing in the test set due to its larger vocabulary. For example, in the Chinese test set, the OOVs rate for NMT system is 4.62%. In the final result of the proposed framework, the OOVs rate is reduced to 2.36%.
- The improvement of translation performance is also attributed to the reordering and correction of phrases. We will carry out human evaluation and look into more details in future.

6.4 Examples

To further analyse the proposed NMT⇒SMT framework, Table 6 shows two examples produced from the baseline NMT system and the corresponding NMT⇒SMT from the JP→EN and ZH→EN tasks, respectively. The first example in Table 6 shows that the SMT system has the capability of making local translations more fluent. We can see that the NMT⇒SMT-B changes the phrase "the hydrogen bond network" in NMT to "hydrogen bond networks" which exactly matches the reference. "NMT-OOV" in the second example indicates the pre-translations after tracking the "UNK" symbols and replacing them by source-side OOVs. This example shows the capability of SMT to make the translation more adequate by subsequently translating the OOV.

Reference:	next , the change of **hydrogen bond networks** which was a basis of the **motion** of the water was explained .
NMT:	next , the change of **the hydrogen bond network** which was a basis of the movement of the water was explained .
NMT⇒SMT-B:	next , the change of **hydrogen bond networks** which was a basis of the movement of the water was explained .
Reference:	**barratt** said : " we have not achieved further information . "
NMT:	**UNK** said : " we have yet to get any results . "
NMT-OOV:	巴拉特 said : " we have yet to get any results . "
NMT⇒SMT-B:	**barratt** said : " we have yet to get any results . "

Table 6: Examples

7 Conclusion

In this paper we propose a cascaded hybrid framework (NMT⇒SMT) to combine NMT and SMT to improve translation performance. More specifically, we first employ a trained NMT system to pre-translate the training data, and then train an SMT system using the pre-translated data. Finally, the tuned target–target SMT system is utilised to re-decode the pre-translated test set and produce the final results. We compare the proposed NMT⇒SMT pipeline with the SMT⇒NMT pipeline on JP–EN and ZH–EN tasks, and show that our framework is more effective than SMT⇒NMT, resulting in improvements on the test set of 2.38 BLEU points and 4.22 BLEU points on JP→EN and ZH→EN, respectively, compared to the baseline NMT system.

As to future work, we expect more experiments on different language pairs and larger-scale data sets to verify the proposed framework, and we will explore better combination of NMT and SMT to further improve translation quality. Additionally, we also want to verify the HMT framework without replacing OOVs in the NMT outputs.

Acknowledgements

We would like to thank the reviewers for their valuable and constructive comments. This research is supported by the ADAPT Centre for Digital Content Technology, funded under the SFI Research Centres Programme (Grant 13/RC/2106), and by SFI Industry Fellowship Programme 2016 (Grant 16/IFB/4490).

References

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, pages 1–15, San Diego, USA.

Bentivogli, L., Bisazza, A., Cettolo, M., and Federico, M. (2016). Neural versus phrase-based machine translation quality: A case study. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 257–267, Austin, Texas, USA.

Chen, Y., Eisele, A., Federmann, C., Hasler, E., Jellinghaus, M., and Theison, S. (2007). Multi-engine machine translation with an open-source decoder for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 193–196, Prague, Czech Republic.

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 1724–1734, Doha, Qatar.

Ding, S., Duh, K., Khayrallah, H., Koehn, P., and Post, M. (2016). The JHU machine translation systems for WMT 2016. In *Proceedings of the Conference on Statistical Machine Translation*, Berlin, Germany.

Enache, R., España-Bonet, C., Ranta, A., and Màrquez, L. (2012). A hybrid system for patent translation. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation*, pages 269–276, Trento, Italy.

Farajian, M. A., Chatterjee, R., Conforti, C., Jalalvand, S., Balaraman, V., Gangi, M. A. D., Ataman, D., Turchi, M., Negri, M., and Federico, M. (2016). FBK's neural machine translation systems for IWSLT 2016. In *Proceedings of the 13th Workshop on Spoken Language Translation*, pages 8–15, Seattle, USA.

Groves, D. and Way, A. (2005). Hybrid example-based SMT: the best of both worlds? In *Proceedings of the Workshop on Building and Using Parallel Texts – Data-driven machine translation and beyond*, pages 183–190, Ann Arbor, USA.

Groves, D. and Way, A. (2006). Hybridity in MT: experiments on the Europarl corpus. In *Proceedings of the 11th Annual Conference of the European Association for Machine Translation*, pages 115–124, Oslo, Norway.

He, W., He, Z., Wu, H., and Wang, H. (2016). Improved neural machine translation with SMT features. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 151–157, Phoenix, Arizona, USA.

Heafield, K., Pouzyrevsky, I., Clark, J. H., and Koehn, P. (2013). Scalable modified Kneser-Ney language model estimation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 690–696, Sofia, Bulgaria.

Jean, S., Cho, K., Memisevic, R., and Bengio, Y. (2015). On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1–10, Beijing, China.

Junczys-Dowmunt, M., Dwojak, T., and Hoang, H. (2016). Is neural machine translation ready for deployment? A case study on 30 translation directions. In *Proceedings of the International Workshop on Spoken Language Translation*, Tokyo, Japan.

Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA.

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

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics on Interactive Poster and Demonstration Sessions*, pages 177–180, Prague, Czech Republic.

Lee, H.-G., Lee, J., Kim, J.-S., and Lee, C.-K. (2015). NAVER machine translation system for wat 2015. In *Proceedings of the 2nd Workshop on Asian Translation (WAT2015)*, pages 69–73, Kyoto, Japan.

Li, H., Zhao, K., Hu, R., Zhu, Y., and Jin, Y. (2015). A hybrid system for Chinese-English patent machine translation. In *Proceedings of MT Summit XV: Sixth Workshop on Patent and Scientific Literature Translation (PSLT6)*, pages 52–67, Miami, USA.

Li, P., Liu, Y., Sun, M., Izuha, T., and Zhang, D. (2014). A neural reordering model for phrase-based translation. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 1897–1907, Dublin, Ireland.

Luong, T., Pham, H., and Manning, C. D. (2015a). Effective approaches to attention-based neural machine translation. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*, pages 1412–1421, Lisbon, Portugal.

Luong, T., Sutskever, I., Le, Q., Vinyals, O., , and Zaremba, W. (2015b). Addressing the rare word problem in neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 11–19, Beijing, China.

Nakagawa, T. (2015). Efficient top-down BTG parsing for machine translation preordering. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference of the Asian Federation of Natural Language Processing*, pages 208–218, Beijing, China.

Nakazawa, T., Yaguchi, M., Uchimoto, K., Utiyama, M., Sumita, E., Kurohashi, S., and Isahara, H. (2016). ASPEC: Asian Scientific Paper Excerpt Corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, Portorož, Slovenia.

Neubig, G., Morishita, M., and Nakamura, S. (2015). Neural reranking improves subjective quality of machine translation: NAIST at WAT2015. In *Proceedings of the 2nd Workshop on Asian Translation (WAT2015)*, pages 35–41, Kyoto, Japan.

Neubig, G., Nakata, Y., and Mori, S. (2011). Pointwise prediction for robust, adaptable japanese morphological analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 529–533, Portland, Oregon, USA.

Neubig, G., Watanabe, T., and Mori, S. (2012). Inducing a discriminative parser to optimize machine translation reordering. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning*, pages 843–853, Jeju Island, Korea.

Niehues, J., Cho, E., Ha, T.-L., and Waibel, A. (2016). Pre-translation for neural machine translation. In *Proceedings of the COLING*, pages 1828–1836, Osaka, Japan.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. (2002). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, USA.

Passban, P., Hokamp, C., and Liu, Q. (2015). Bilingual distributed phrase representations for statistical machine translation. In *Proceedings of MT Summit XV*, pages 310–318, Miami, USA.

Paul, M., Doi, T., Hwang, Y., Imamura, K., Okuma, H., and Sumita, E. (2005). Nobody is perfect: ATR's hybrid approach to spoken language translation. In *Proceedings of the International Workshop on Spoken Language Translation: Evaluation Campaign on Spoken Language Translation*, Pittsburgh, USA.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Barone, A. V. M., Mokry, J., and Nădejde, M. (2017). Nematus: a toolkit for neural machine translation. In *arXiv:1703.04357*.

Sennrich, R. and Haddow, B. (2016). Linguistic input features improve neural machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 83–91, Berlin, Germany.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725, Berlin, Germany.

Sutskever, I., Vinyals, O., , and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 2014 Neural Information Processing Systems*, pages 3104–3112, Montreal, Canada.

Sánchez-Martínez, F., Pérez-Ortiz, J. A., and Forcada, M. L. (2007). Integrating corpus-based and rule-based approaches in an open-source machine translation system. In *Proceedings of the METIS-II Workshop: New Approaches to Machine Translation*, Leuven, Belgium.

Toral, A. and Sánchez-Cartagena, V. M. (2017). A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, Valencia, Spain.

Tu, Z., Liu, Y., Shang, L., Liu, X., and Li, H. (2017). Neural machine translation with reconstruction. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, San Francisco, USA.

Wang, X., Lu, Z., Tu, Z., Li, H., Xiong, D., and Zhang, M. (2017). Neural machine translation advised by statistical machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, San Francisco, California, USA.

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. In *arXiv:1609.08144*.

Zeiler, M. D. (2012). ADADELTA: An adaptive learning rate method. In *CoRR, abs/1212.5701*.

Zhang, H., Yu, H., Xiong, D., and Liu, Q. (2003). HHMM-based chinese lexical analyzer ICTCLAS. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 184–187, Sapporo, Japan.

Zhao, Y., Huang, S., Chen, H., and Chen, J. (2014). Investigation on statistical machine translation with neural language models. In *Proceedings of Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, pages 175–186.