A Smorgasbord of Features to Combine Phrase-Based and Neural Machine Translation

Benjamin Marie bmarie@nict.go.jp
Atsushi Fujita atsushi.fujita@nict.go.jp
National Institute of Information and Communications Technology,
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan

Abstract

Superiority of neural machine translation (NMT) and phrase-based statistical machine translation (PBSMT) depends on the translation task. For some translation tasks, such as those involving low-resource language pairs or close languages, NMT may underperform PBSMT. In order to have a translation system that performs consistently better regardless of the translation task, recent work proposed to combine PBSMT and NMT approaches. In this paper, we propose an empirical comparison of the most popular existing approaches that combine PBSMT and NMT. Despite its simplicity, our simple reranking system using a smorgasbord of informative features significantly and consistently outperforms other methods, even for translation tasks where PBSMT and NMT produce translations of a very different quality.

1 Introduction

Neural machine translation (NMT) systems have been shown to outperform phrase-based statistical machine translation (PBSMT) systems in many translation tasks. NMT systems perform especially well with language pairs involving two distant languages or morphologically-rich languages. Translations produced by NMT systems are usually more fluent than those produced by state-of-the-art PBSMT systems. However, NMT systems are still far from producing perfect translations. Many researchers have studied the weaknesses of the NMT approach and shown that NMT systems perform poorly compared to PBSMT systems in relatively common scenarios, especially those involving low-resource language pairs (Bentivogli et al., 2016; Koehn and Knowles, 2017). Several approaches have recently been proposed to combine PBSMT and NMT in order to exploit their complementarity and to produce better translations.

In this paper, we study the most popular combination methods and empirically compare them, aiming at drawing a better picture of their strengths and weaknesses. We demonstrate that reranking the simple concatenation of n-best lists produced by each of the NMT and PBSMT systems, with a set of well-motived features, performs consistently the best compared to the more popular and complex methods proposed by previous work. We also show that, while other approaches can perform worse than the best system, a simple reranking approach offers some guarantee that the selected best translation will be rarely worse than the best one proposed by the PBSMT or by the NMT system, even when one of the systems performs very poorly.

The remainder of this paper is organized as follows. In Section 2, we review the existing methods used to combine PBSMT and NMT. Then, in Section 3, we make our assumption that a reranking system using a large set of informative features can outperform other existing methods. We evaluate our proposed reranking systems in Section 4 and analyze our results in Section 5. In Section 6, we conclude and propose promising perspectives for this work.

2 Current Approaches to Combine PBSMT and NMT

This section reviews four different methods able to combine PBSMT and NMT: confusion network decoding (Section 2.1), pre-translation with a PBSMT system (Section 2.2) and rescoring PBSMT or NMT translation hypotheses using different models (Section 2.3 and Section 2.4). We do not include in our comparison the work of He et al. (2016), which uses SMT features during NMT decoding, because their method cannot use phrase translation probability or more complex models that cannot be used during decoding. Moreover, we leave for future work the study of the more recent method proposed by Zhou et al. (2017), which combines MT system outputs using neural networks. This method outperformed confusion network decoding, but has been evaluated only on a Chinese-to-English translation task with PBSMT and NMT systems that performed comparably on this task.

2.1 Confusion Network Decoding

The first application of machine translation (MT) system combination used a consensus decoding strategy relying on a confusion network (Bangalore et al., 2001). Since this first work, this approach has been improved and remains one of the most popular methods to combine many translations produced by different MT systems (Freitag et al., 2014).

To generate the confusion network, alignments are required between the tokens of all the translation hypotheses to combine. Previous work (Heafield and Lavie, 2011; Freitag et al., 2014) on system combination used METEOR (Denkowski and Lavie, 2014) to perform an accurate word alignment between translation hypotheses by making use of its ability to align synonyms, stems, and paraphrases. After building the confusion network, decoding is performed to find the most consensual path with additional models such as a large language model.

This approach finds usually a better translation hypothesis than the best translations produced by the individual systems. However, it becomes quickly prohibitive if one wants to combine hundreds of hypotheses, such as the n-best hypotheses generated by different systems, while using costly models to score the decoding paths. Moreover, we have no guarantee that the output of the combination will be better than the best hypothesis generated by individual systems. The confusion network may allow the generation of many hypotheses of very poor quality, especially in cases where many of the translation systems perform much worse than the best systems used in the combination.

2.2 Pre-translation with a PBSMT system

Pre-translation is a recent method dedicated to combine PBSMT and NMT in a simple pipeline (Niehues et al., 2016). First, a PBSMT system is trained and used to decode the source side of the training data of the NMT system. Then, a second-stage NMT system is trained, where the concatenation of the source sentence and the PBSMT-decoded translation is regarded as the new source side of training data, while the target side of the training data remains unchanged.

The main motivation behind this work is that a pre-translation generated by a PBSMT system would be informative to better guide the training of NMT systems. However, as suggested by Niehues et al. (2016), to improve an NMT system with a pre-translation, the PBSMT system must produce translations of a quality comparable to (or better than) those produced by the NMT system. In cases where the PBSMT system produces translation of poor quality, we can expect that such pre-translation will significantly harm the training of the NMT system.

2.3 Rescoring PBSMT hypotheses with NMT

Before the emergence of end-to-end NMT systems, it was a common practice to include neural network models in PBSMT for reranking the n-best translation hypotheses produced by the PBSMT system (Le et al., 2012) or to include them directly during decoding (Devlin et al.,

2014; Junczys-Dowmunt et al., 2016). This strategy has been successfully exploited for PBSMT systems. However, it is currently less attractive, because NMT systems are often able to produce much better translations than PBSMT systems, even better than the best translations obtained after reranking the PBSMT system's n-best hypotheses with NMT system's models.

2.4 Phrase-based Forced Decoding

Yet another recent method dedicated to combine PBSMT and NMT systems is called *phrase-based forced decoding* (Zhang et al., 2017) (henceforth, PBFD). The idea is to use the phrase table and its translation probabilities, which are commonly learned during the training of a PBSMT system, to rescore translations produced by an NMT system.

This approach aims at alleviating the low adequacy of some of the translations produced by an NMT system. Since this approach relies directly on the phrase table usually used in PBSMT, it will promote hypotheses that matches phrase pairs associated with a high translation probability from the phrase table. The forced decoding searches for the best phrase-based segmentation and returns the corresponding phrase-based translation probability.

PBFD is extremely costly to perform during NMT decoding but rather feasible after it on a selected set of diverse hypotheses. Then, given the PBFD score and the original score given by the NMT system, the rescoring of the hypotheses is performed. Zhang et al. (2017) did not rerank n-best lists but instead reranked a sample of hypotheses extracted from the NMT decoder's search space. This amplifies the diversity among the hypotheses to rescore, and the increased diversity has been shown useful in training a reranking system (Gimpel et al., 2013). However, as a potential drawback, hypotheses of very bad quality could be chosen.

3 n-best List Reranking

3.1 n-Best List Combination

Since the early age of PBSMT (Och et al., 2004), reranking the n-best lists of hypotheses produced by a PBSMT system has been shown to be a simple and efficient way to use complex features that could not be used during decoding. Furthermore, this approach offers some good guarantee to find a better translation, because rescoring is applied to the best part of the decoder's search space, while making use of more, and potentially better, features than the decoder. However, unlike pre-translation or confusion network decoding approaches, a simple reranking of the hypotheses produced by a single decoder, NMT or PBSMT, is limited in its ability to take advantage of the complementarity of both approaches. For instance, if an NMT system produces fluent but inadequate n-best translations, a simple reranking of this n-best list with PBSMT models can only help to find an hypothesis which is less inadequate. Reranking NMT n-best hypotheses does not give access to the PBSMT decoder's search space and its potentially more adequate translations.

Instead of a list of hypotheses produced by a single system or multiple but homogeneous systems, we merge two lists respectively produced by PBSMT and NMT decoders, and rescore all the hypotheses. Then, the reranking framework using a lot of features to better model the fluency and the adequacy of the hypotheses can potentially find a better hypothesis than the one-best hypotheses originated by either the PBSMT or NMT systems. This method is similar to the one proposed by Hildebrand and Vogel (2008). However, their work aims at combining n-best lists from any kind of MT systems, ignoring the specificities and models of the systems used to produce them. In contrast, we focus on PBSMT and NMT system combination by making use of their respective models. This method has never been evaluated in comparison with the state-of-the-art methods presented in Section 2.

While this approach seems simple, mixing efficiently both kinds of hypotheses is actually challenging. For instance, if we choose only the model scores from an NMT system as features,

it is likely that all PBSMT hypotheses will be ignored by the reranking framework by giving a high preference to the hypotheses with high NMT models score, which will be actually the ones produced by the NMT system.

3.2 Reranking Framework and Features

Previous work on n-best list reranking has proposed many different training algorithms, including those used to optimize PBSMT systems, such as MERT (Och, 2003) and KB-MIRA (Cherry and Foster, 2012). We choose KB-MIRA since it is commonly used in reranking framework and provides stable performances. It can also handle many features as opposed to MERT.

The features we used are commonly used for n-best list reranking, which are difficult or impossible to use during NMT or PBSMT decoding. To the best of our knowledge, the following features have never been exploited together in the same reranking framework.

3.2.1 NMT Features

NMT translation models can be used to score a translation produced by an arbitrary system. We only need the source sentence and the corresponding translation hypothesis. These models have been used to rerank n-best lists of hypotheses produced by PBSMT systems and can also be used to rescore hypotheses produced by other NMT systems. Different NMT translation models, generated at different training epochs, or by independent training runs, can be combined to make an ensemble of models to better score translation hypotheses.

right-to-left NMT translation models, trained on parallel data in which the target side sequences of tokens are reversed, are also useful. Such *right-to-left* models have shown good performance in reranking n-best lists of hypotheses (Sennrich et al., 2017a).

3.2.2 PBSMT Features

A state-of-the-art PBSMT system uses the log-linear combination of several models:

- a phrase table containing phrase pairs associated with a set of translation probabilities, which controls the adequacy of the translation
- a language model controlling the fluency of the translation
- a distortion score that controls how much the target phrases in the translation hypothesis have been reordered given their corresponding source phrases
- a lexical reordering table to control three kinds of phrase-based reordering: monotonous, swap, or discontinuous (henceforth, MSD models)
- a word penalty to penalize short translations
- a phrase penalty to count the number of phrase pairs used to compose the translation

While translation models (Zhang et al., 2017) and language models (Wang et al., 2017) are useful to rescore NMT hypotheses, this may not be the case for the reordering models. A state-of-the-art PBSMT decoder limits its search within a pre-determined distortion limit. This limit can be seen as a safeguard to prevent the decoder to generate very ungrammatical translations, since it does not have the ability to model long dependencies between tokens. In contrast, NMT decoders are free to perform long-distance reorderings. For language pairs that need long-distance reordering, this means that an NMT hypothesis of a good quality will have a high distortion score and many source phrases translated discontinuously. The PBSMT reordering models seem then inadequate to score NMT hypotheses in our reranking framework, especially since we will keep using NMT models that already model the fluency.

We perform PBFD on the NMT hypotheses (Section 2.4) using a PBSMT system's phrase table, and use the score produced by PBFD as a feature. For the PBSMT hypotheses, we use directly the phrase segmentation produced by the PBSMT system and compute the same score.

It is also possible to use the full PBSMT system's scoring function to score NMT hypotheses. Indeed, PBFD splits the NMT hypothesis into phrase pairs. Then, we can further exploit this segmentation to compute all PBSMT features and combine them log-linearly using the same model weights found during the tuning of the PBSMT system. Nonetheless, PBFD generates a phrase segmentation that may be unreliable to compute all PBSMT model scores, especially because most of the NMT hypotheses may be unreachable by a PBSMT system, leaving some source and target tokens out of a phrase pair.

3.2.3 Sentence-Level Translation Probability

While the PBFD uses only phrase translation probabilities, it is often a good idea to use also lexical translation probabilities in order to get a smoothed score. Since an NMT system does not produce word alignments, we consider to take the average of the lexical translation probabilities over all possible word pairs between the source sentence f and the translation hypothesis e, according to the following formula:[1]

$$P_{avg}(e|f) = \frac{1}{I}\sum_{i=1}^{I}\log\Big(\frac{1}{J}\sum_{j=1}^{J}p(e_i|f_j)\Big) \tag{1}$$

where I and J are the lengths of e and f, respectively, and $p(e_i|f_j)$ the lexical translation probability of the i-th target word e_i of e given the j-th source word f_j of f. Since Equation (1) is dominated by the highest lexical translation probability, Hildebrand and Vogel (2008) also proposed to compute the translation probability given by the following equation:

$$P_{lmax}(e|f) = \frac{1}{I}\sum_{i=1}^{I}\log\Big(\max_{j}p(e_i|f_j)\Big) \tag{2}$$

As the features for rescoring, we compute the scores given by Equations (1) and (2) for both translation directions using the lexical translation probabilities trained on the same parallel data used to train the MT systems.

3.2.4 Word Posterior Probability

Word posterior probability (WPP) is another feature that is commonly used in PBSMT to rerank lists of translation hypotheses. For all target tokens appearing in the list, it computes the probability for the token to appear in a translation hypothesis. Then, we can score an entire hypothesis by averaging the posterior probability of the tokens it contains. We use the count-based WPP as defined by Ueffing and Ney (2007). WPP is computed given the decoder's score of the hypotheses in which the word appears. Since our list of hypotheses to rerank contains hypotheses produced by two different decoders, we compute two different WPP: one based on the score computed by Equation (1), with direct translation probabilities, and the other based on the score computed by the NMT decoder.

3.2.5 Consensus Score

The so-called minimum Bayes risk (MBR) decoding for n-best list is a popular method used in SMT to find in an n-best list of hypotheses the one that is on average the most similar to the other hypotheses. Sentence-level BLEU (Papineni et al., 2002) (sBLEU) is usually considered as the metric used to measure the similarity between hypotheses (Ehling et al., 2007).

This method has a common objective with confusion network decoding and WPP (Section 3.2.4), since we search for the hypothesis containing the most popular tokens or n-grams used by the decoder to construct its n-best hypotheses.

[1] Applying forced word alignment on the NMT hypotheses would be an alternative, but we did not observe any significant differences in our preliminary experiments.

We gauge how each hypothesis is similar to all the other hypotheses, using two scores respectively based on *s*BLEU and chrF++ (Popović, 2017).

3.2.6 Other Features

Depending on the origin of the hypothesis, generated either by PBSMT or NMT systems, some features can give significantly different scores. To help our reranking system to weight these differences, we introduce a binary feature that only indicates whether the hypothesis has been produced by a PBSMT or an NMT system. Regular attention-based NMT systems have no direct mechanism to control the length of the hypotheses produced, but information about the hypothesis length can help to improve the performance (Zhang et al., 2017). In addition to the word penalty used by the PBSMT system, we also add the difference between the number of tokens in the source sentence and that for the translation hypothesis, and its absolute value.

4 Experiments

4.1 Data

We conducted experiments on two significantly different language pairs: Japanese–English (Ja-En) and French–English (Fr-En). Ja-En involves two distant languages for which an NMT decoder is expected to perform much better than a PBSMT decoder, especially due to the long-distance reordering to perform to get a good translation. In contrast, Fr-En involves much closer languages with usually only local reorderings to perform. We thus expect PBSMT and NMT to provide more similar results given a large set of parallel data.

For Ja-En, we used the NTCIR Patent Translation Task (Goto et al., 2013). We used the parallel data provided for the task to train PBSMT and NMT systems. The language models for Japanese and English were trained on the target side of the parallel data and the entire NTCIR monolingual data. The NTCIR development data were used as a validation dataset during the training of the NMT system and to tune the PBSMT system. We used the NTCIR-9 test (`T09`) and NTCIR-10 test (`T10`) for evaluation. For Fr-En, we used data provided for the WMT'15 News Translation Task.[2] Our parallel data used to train the systems comprise Europarl v7, 10^9 French–English, and news-commentary v10. The language models for French and English were trained on the target side of the parallel data and the entire News Crawl corpora. We used newstest2012 as a validation dataset during the training of the NMT system and to tune the model weights of the PBSMT system, and newstest2013 (`N13`) and newstest2014 (`N14`) for evaluation. The statistics of the data are presented in Table 1.

4.2 Baseline Systems

To train and test our PBSMT systems and attention-based NMT systems, we respectively used `Moses` (Koehn et al., 2007) and `Nematus` (Sennrich et al., 2017b) frameworks.

For our baseline PBSMT systems, word alignments were trained with `mgiza` and `fast_align` (Dyer et al., 2013) respectively for Ja-En and Fr-En.[3] After their training for both translation directions, word alignments are symmetrized using the `grow-diag-final-and` heuristic. We trained two 4-gram language models with `lmplz` (Heafield et al., 2013) for each translation direction, one trained on the target side of the parallel data, and the other on the monolingual data concatenated to the target side of the parallel data. We pruned all singletons for the Japanese and English second language models used for the NTCIR translation tasks, because the monolingual data are very large. The reordering models are MSD lexicalized and

[2] `http://www.statmt.org/wmt15/translation-task.html`

[3] We did not use `mgiza` to train the word alignments for the Fr-En pair, since `fast_align` is much more efficient on large training data, while it has been shown to perform as well as `mgiza` for this language pair.

Datasets		#sentences	#tokens			#token types		
			Ja	Fr	En	Ja	Fr	En
NTCIR	parallel	3M	110M		102M	169k		275k
	development	2,000	73k		67k	4k		5k
	T09$_{Ja\rightarrow En}$	2,000	74k		68k	5k		6k
	T09$_{En\rightarrow Ja}$	2,000	74k		70k	5k		6k
	T10$_{Ja\rightarrow En}$	2,300	99k		92k	6k		7k
	T10$_{En\rightarrow Ja}$	2,300	87k		80k	6k		6k
	monolingual	-	27B		15B	9M		22M
WMT	parallel	24M		726M	614M		2M	2M
	development	3,003		82k	73k		11k	10k
	N13	3,000		74k	70k		11k	9k
	N14	3,003		81k	71k		11k	10k
	monolingual	-		2B	3B		4M	6M

Table 1: Statistics on train, development, and test data.

bidirectional models. PBSMT systems are tuned with KB-MIRA using development data. The distortion limit was tuned and set to 16 for Ja-En.[4]

Our baseline NMT systems used the default training parameters of Nematus, with layer normalization, and performed BPE (Sennrich et al., 2016) to fix the source and target vocabulary sizes at 50k. The BPE segmentation was jointly learned for French and English since they share the same alphabet. During Nematus training, we saved the model after every 5k mini-batch iterations. The 4-best models according to their performance on the development data were selected to perform ensemble decoding. The decoding were performed using a beam size of 100 to produce 100-best hypotheses. As suggested by Koehn and Knowles (2017), we normalized the hypothesis score by their length during decoding to prevent a drop of the NMT system performance when using such a large beam size.

For system combination with confusion network decoding (Section 2.1), we used the Jane framework (Freitag et al., 2014). We evaluated two systems: one combining only the one-best hypotheses produced by Moses and Nematus ($n = 1$), and the other combining all the hypotheses in the 100-best lists ($n = 100$). During decoding, we used all the default models in addition to the large language models that were used by our PBSMT baseline systems.[5]

For pre-translation (Section 2.2), we decoded our entire training data with our PBSMT system and concatenated each of the results to its source side to train a second-phase NMT system (henceforth, Pre-Nematus), exactly as described in (Niehues et al., 2016). To evaluate Pre-Nematus, we performed ensemble decoding using the 4-best models.

We also evaluated two baseline reranking systems: one using NMT models to rerank the PBSMT system's n-best list (Section 2.3), denoted R$_{nmt}$, and the other using the PBFD features (Section 2.4), denoted R$_{pbsmt}$. R$_{nmt}$ uses all the features used by the Moses scoring function, the Moses score, the 4 translation models used by the Nematus baseline system, and the 4-best right-to-left translation models (Section 3.2.1). For R$_{pbsmt}$, we used the same features as described by Zhang et al. (2017): the scores given by Nematus and the 4 best left-to-right models, the score given by the PBFD, and the word penalty (as defined in Moses).

R$_{nmt}$ reranks the n-best list of distinct hypotheses (M) produced by Moses, and R$_{pbsmt}$ reranks the n-best list of hypotheses (N) produced by Nematus.

[4]The default value in Moses is not appropriate for distant languages.

[5]We could have also reranked the n-best lists produced by Jane. However, we found out that Jane's n-best lists produced for system combination are of a very poor quality and not diverse enough to train a reranking system.

Feature	Description
L2R (5)	The scores given by each of the 4-best left-to-right Nematus models and their geometric mean
R2L (5)	The scores given by each of the 4-best right-to-left Nematus models and their geometric mean
PBFD (1)	The PBFD score (Section 2.4)
LEX (4)	The sentence-level translation probabilities (Section 3.2.3), computed using Equations (1) and (2), for both translation directions
LM (2)	Scores given by the two language models used by the Moses baseline systems
WP (1)	Word penalty
TM (4)	PBSMT translation model scores computed according to the probabilities given by the Moses phrase table on the phrase segmentation produced by Moses for the hypotheses in M, or by the PBFD for the hypotheses in N (same segmentations were used to compute scores with MSD models (MSD), the distoration score (DIS), the phrase penalty (PP), and the Moses score (MOSES))
MSD (6)	Scores computed using the Moses MSD lexical reordering table
DIS (1)	Distortion score
PP (1)	Phrase penalty
MOSES (1)	Score given by Moses for the hypotheses in M. For the hypotheses in N, we compute this score using all the Moses models and their weights
WPP (2)	Word posterior probability (Section 3.2.4)
MBR (2)	MBR decoding applied on M+N (Section 3.2.5), using sBLEU and chrF++
LEN (2)	The difference between the length of the source sentence and the length of the translation hypothesis, and its absolute value
SYS (1)	System flag, 1 if the hypothesis is in M or 0 if it is in N

Table 2: Set of features used by $\texttt{R}_{full}$. $\texttt{R}_{sub}$ uses only the features in bold. The numbers between parentheses indicate the number of scores in each feature set.

4.3 Reranking Systems

We trained and evaluated our reranking systems using two different sets of features to rerank the concatenation of M and N (henceforth, M+N). Our first system, denoted $\texttt{R}_{full}$, used the full set of 38 features described in Section 3.2 and listed in Table 2. Our second reranking system, denoted $\texttt{R}_{sub}$, used only a subset of the features. We excluded most of the phrase-based features, considering that they are unreliable to score NMT hypotheses. The impact of each feature is analyzed in Section 5.1. All our reranking systems were trained with KB−MIRA on n-best lists produced for the development data.[6] For a comparison, we also evaluated both reranking systems with M and N reranked separately. For each reranking experiments, both training and testing n-best lists were generated by the same system.

4.4 Results

As shown in the first two rows of Table 3, the superiority of NMT and PBSMT depends on the language pair. As expected, for the Ja-En pair, NMT performed much better than PBSMT. For both test sets, T09 and T10, the large difference between Moses and Nematus reached approximately 10 BLEU points. In contrast, for the Fr-En pair, PBSMT performed better than NMT, slightly on N13 and significantly on N14. The difference in translation quality between both systems makes it challenging for a system combination to perform consistently better than the best single system, regardless of the translation task.

Jane did not improve the translation quality over Nematus for Ja→En. We observed improvements for the other three tasks when combining the one-best hypotheses produced by Moses and Nematus. However, when we combined the 100-best hypotheses for the Ja-En

[6] We used the rescoring implementation provided by Moses.

Configuration	Ja→En		En→Ja		Fr→En		En→Fr	
	T09	T10	T09	T10	N13	N14	N13	N14
Moses	31.3	32.7	34.7	35.9	31.4	39.9	30.6	39.1
Nematus	41.9	41.6	44.8	45.4	30.8	34.0	30.6	35.8
Jane M+N (n=1)	41.5	41.6	44.9	45.8	32.0	40.0	30.8	39.5
Jane M+N (n=100)	39.0	40.2	41.6	42.7	32.1	40.0	31.0	39.9
Pre-Nematus	31.4	29.7	33.5	33.8	30.0	37.2	29.5	37.4
R_{nmt} M	33.3	34.1	36.8	38.3	33.6	41.4	32.4	40.5
R_{pbsmt} N	42.5	43.1	46.1	46.7	32.5	34.7	31.4	36.1
R_{full} M	33.4	34.2	36.7	38.4	33.6	41.4	32.4	40.5
R_{full} N	42.5	43.6	46.3	47.1	33.5	37.6	32.4	38.8
R_{full} M+N	42.3	43.7	46.3	47.0	33.8	41.4	32.5	40.6
R_{sub} M	33.5	34.2	36.7	38.5	33.6	41.4	32.4	40.4
R_{sub} N	43.0	43.8	46.9	47.5	33.9	38.0	32.3	38.8
R_{sub} M+N	43.0	43.9	47.1	47.7	34.2	41.6	32.6	40.8

Table 3: Results (BLEU) produced by the baseline systems and our reranking systems respectively presented in Section 4.2 and Section 4.3.

pair, the performance dropped significantly, probably due to the low quality of the hypotheses produced by Moses. As for Pre-Nematus, for all tasks we did not manage to obtain improvement over the best single system. The produced translations were much worse for Ja-En, especially on T10 compared to Nematus with a drop of 11.9 and 11.6 BLEU points respectively for the Ja→En and En→Ja tasks. Nematus was potentially more disturbed than helped by the very low quality of the translations provided by Moses. For Fr-En, we did not observe such a drop, probably due to the much better quality of the PBSMT translations, but neither got any improvements over Moses. However, we could observe significant improvements over Nematus on N14, of up to 3.2 BLEU points for Fr→En, showing that a pre-translation of a good quality can significantly help the NMT system. Both Jane and Pre-Nematus provided inconsistent results in our translation tasks and underperformed when the difference between the PBSMT and NMT system translation quality was very large.

R_{nmt} and R_{pbsmt} performed significantly better than the system that produced the n-best list they reranked. The reranking system R_{nmt} M gave the best results for the tasks where PBSMT performed the best, with up to 2.2 BLEU points of improvements on Fr→En N13. In contrast, R_{pbsmt} N performed the best for Ja-En, with for instance a surprising 1.5 BLEU points improvement on Ja→En T10. Despite the large difference in translation quality between Moses and Nematus, the PBSMT models seem to be helpful to rerank the n-best lists produced by Nematus. These reranking systems produced better results than the best single system. However, they can be improved by combining n-best lists and by using more features to perform a more informed reranking.

Indeed, R_{full} M+N consistently performed similarly or better in all the four translation tasks. Reranking the concatenation of Moses and Nematus n-best lists with a set of features derived from NMT and PBSMT models significantly helped to obtain consistent results across our translation tasks, even for Ja-En for which the n-best lists produced by Moses and Nematus were of a very different quality (Section 5.3). However, as forecasted in Section 3.2, using all the features did not give the best results. Removing the phrase-based features, except for the PBFD score, gave better results, especially for the Ja-En pair, with for instance 0.8 BLEU points of improvement on En→Ja T09 obtained by R_{sub} over R_{full}. Over the best single system (Moses

Feature set removed	Ja→En T09		En→Ja T09		Fr→En N13		En→Fr N13		Computational time (ms)
	R_{full}	R_{sub}	R_{full}	R_{sub}	R_{full}	R_{sub}	R_{full}	R_{sub}	
none	42.3	43.0	46.3	47.1	33.8	34.2	32.5	32.6	-
L2R	42.6	43.0	46.3	47.1	**33.6**	**34.1**	**32.2**	**32.5**	1,560
R2L	42.4	**42.6**	46.4	**46.6**	**33.5**	**33.7**	**31.9**	**32.1**	1,890
PBFD	42.6	43.2	46.5	**47.0**	33.8	34.3	**32.4**	**32.3**	240,502
LEX	42.5	43.0	**46.2**	47.1	33.8	**34.1**	32.5	**32.4**	213
LM	42.5	43.1	46.7	47.1	**33.5**	**34.0**	**32.1**	**32.2**	98
WP	42.5	43.1	46.3	47.1	33.8	**34.1**	32.6	32.6	< 1
TM	42.8	-	46.3	-	33.8	-	32.5	-	240,504
MSD	42.5	-	46.5	-	**33.5**	-	32.7	-	240,532
DIS	42.4	-	46.4	-	33.9	-	32.5	-	240,503
PP	42.4	-	46.3	-	33.8	-	**32.3**	-	240,502
MOSES	42.4	-	**46.2**	-	**33.7**	-	32.5	-	240,503
WPP	42.4	43.1	46.3	47.1	33.8	**34.1**	**32.4**	**32.4**	20
MBR	42.8	43.1	46.4	**46.9**	**33.6**	**33.9**	**32.3**	**32.4**	111,232
LEN	42.6	43.1	46.4	47.1	**32.6**	**32.7**	**31.9**	**32.0**	< 1
SYS	42.5	43.1	46.4	47.1	33.8	**34.1**	32.6	32.6	< 1
L2R+R2L	42.4	**42.5**	**46.2**	**46.7**	**33.4**	**33.7**	**32.0**	**32.0**	3,450
PBFD+LEX	42.6	43.1	46.3	**47.0**	**33.6**	34.3	**32.4**	**32.5**	240,715
WPP+MBR	42.7	43.2	46.3	**46.9**	**33.6**	**34.0**	**32.2**	**32.4**	111,252
WP+LEN	42.7	43.0	46.8	**46.7**	**32.6**	**32.8**	**32.0**	**31.9**	1
L2R+R2L+ PBFD+LEX	42.4	**42.4**	**46.1**	**46.2**	**33.4**	**33.6**	**31.9**	**31.8**	244,165

Table 4: Results (BLEU) of the reranking systems R_{full} and R_{sub} obtained after removal of each feature set, independently. Reranking is performed using M+N lists of hypotheses. Bold indicates a deteriorated BLEU score when removing the feature set. The last column indicates the approximate average computational time, needed to compute the feature set, per source sentence (200 hypotheses) for Ja→En T09, using one CPU thread (Xeon E5-2670 2.6 GHz) or one GPU (GeForce GTX 1080), for the features L2R and R2L. Note that for computing a phrase-based feature, we need to first perform PBFD.

or Nematus), our best system, R_{sub}, achieved improvements ranging from 1.1 BLEU points on Ja→En T09 to 2.8 BLEU points on Fr→En N13.

5 Analysis

5.1 Impact of the Features

We evaluated the impact of the features used by our reranking systems, R_{full} and R_{sub}, by removing them, individually or in a subset, during training and testing. The results are presented in Table 4 for the Ja↔En T09 and Fr↔En N13 tasks.

These experiments show that removing features individually has a limited impact on the performance, and many of the features are correlated. Surprisingly, removing all the features based on NMT models and translation probabilities (L2R+R2L+PBFD+LEX) had a relatively limited impact on the performance, with at most a drop of 0.9 BLEU points for En→Ja, while this set of features is also very costly to compute.

Furthermore, we can see that the importance of the features depends on the language pair (and potentially the domain). Removing the language model scores had no impact for Ja↔En

	Ja→En T09			Fr→En N13		
	avg. sBLEU	avg. chrF++	#token types	avg. sBLEU	avg. chrF++	#token types
M	66.1	81.3	6,607	59.5	74.2	13,016
N	65.9	79.8	7,903	58.7	73.4	16,481
M+N	52.5	72.9	8,810	50.7	67.8	18,756

Table 5: Diversity of the hypotheses in the list M, N and M+N.

but consistently decreased the BLEU scores for Fr↔En. Removing LEN led to a significant drop of the BLEU scores for Fr↔En (up to -1.5 BLEU points), while it had no impact on the results for Ja↔En. Left-to-right Nematus models seems to have a limited impact on the results compared to the right-to-left models.

Moreover, as expected, removing the phrase-based features, such as TM and MSD, from R_{full} often improved the performance, due to the unreliability of the phrase segmentation produced by the PBFD on Nematus hypotheses.

5.2 Diversity of n-Best Hypotheses

As pointed out by Gimpel et al. (2013), a high diversity in the lists of hypotheses to rerank, especially in the list used to train the reranking system, is an important criterion to obtain a good performance. We evaluated the diversity of the hypotheses in M, N and M+N, using three indicators: average sBLEU, average chrF++, and the number of token types in the lists. The average sBLEU and the average chrF++ were computed from the MBR feature set, i.e., the average of the MBR scores given all the hypotheses in the list. A lower average sBLEU or chrF++ means that the list contains more diverse hypotheses. These indicators for the Ja→En T09 and Fr→En N13 translation tasks are presented in Table 5.

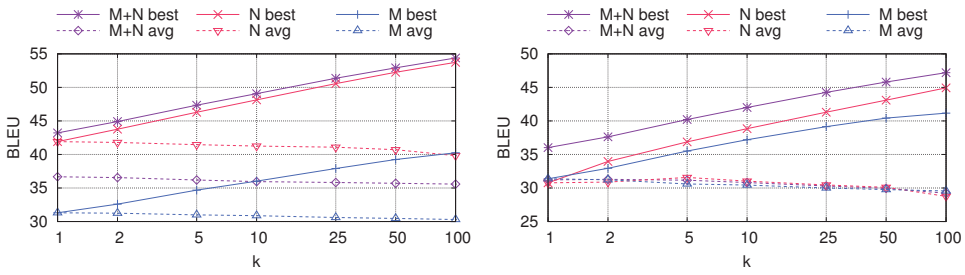
According to sBLEU and chrF++, N seems slightly more diverse than M for both language pairs. N also involved more diverse token types. For instance, within the 100-best hypotheses for the Ja→En T09 translation task, it used 1,296 more types of tokens than Moses. While M and N had almost a similar diversity, their concatenation, M+N, was much more diverse for both language pairs. For the Ja→En T09 translation task, the average sBLEU decreased from 66.1 and 65.9 points, respectively for M and N, to 52.5 points for M+N. This means that the PBSMT and NMT systems tend to produce different sets of translation hypotheses from each other.

As for the origin of the best hypotheses selected from M+N by our best ranking system, R_{sub}, for instance for the Ja→En T09 translation task, 53.5% and 46.6% were respectively those produced by Nematus and Moses. The high ratio of Moses hypotheses may seem surprising given their poor quality for this task. Actually, most of the Moses hypotheses chosen by R_{sub} are similar, or as poor as, the hypotheses produced by Nematus. We will show in the next section that for the Ja→En T09 task, given the low quality of M, this is a very safe choice and that we cannot hope to obtain large improvements by selecting hypotheses from M that are very different from the hypotheses in N.

5.3 Quality of n-Best Hypotheses

In the previous section, we highlighted a high diversity in the list of hypotheses, especially in M+N, which is advantageous to train a reranking system. However, to improve the translation quality with a reranking system, we also need lists of hypotheses of good quality that contain better hypotheses than the best output of the PBSMT and NMT systems.

We analyzed the quality of n-best lists using two indicators: an *oracle best* and an *oracle average*. For each source sentence, the former finds the translation hypothesis in the list that has the highest sBLEU score, given the reference translation, and then a standard BLEU score over



(a) Ja→En T09 (b) Fr→En N13

Figure 1: The oracle BLEU scores computed on M, N and M+N. The k-best hypotheses of the 100-best lists are used to compute the oracle.

such hypotheses in the test set is computed. To compute the oracle average scores, we used the same strategy, but selected hypotheses that had the closest sBLEU score to the average sBLEU score given the hypotheses in the list. We performed these oracle experiments on the Ja→En T09 and Fr→En N13 translation tasks. The results are presented in Figure 1.

As expected, we faced two very different situations. For Ja→En, the oracle best for k=100 computed on the list M generated by Moses did not even reach the BLEU score of the Nematus translation at k=1. Moreover, concatenating the entire M and N improved the oracle best scores only slightly, with less than one BLEU points of improvement for k=100 compared to using only N. Despite this large difference in quality, as we saw in Section 4.4, the concatenation was not harmful and the features were informative enough to help the reranking system. In contrast, for Fr→En M and N seemed much more complementary, as their concatenation improved the oracle best score of more than two BLEU points at k=100. We also observed that M and N for this translation task had very similar oracle average scores, while the concatenation of them did not decrease the oracle average score of the list.

6 Conclusion

We presented a simple reranking system guided by a smorgasbord of diverse features and showed that it can significantly outperform the state-of-the-art methods that combine PBSMT and NMT. Our reranking system managed to put at the first rank better translation hypotheses than the one-best hypotheses found by each of the PBSMT and NMT systems, relying on the diversity and quality of their respective n-best lists. Moreover, we demonstrated that our reranking system has the ability to perform consistently in two different configurations, even when the component systems produced translations of a very different quality.

As future work, we plan to study whether a reranking system can also improve the translation quality in low-resource conditions. Indeed, in this situation, PBSMT performs much better than NMT. It will thus be worth seeing whether our framework can help to identify some NMT hypotheses that are better than PBSMT hypotheses.

Acknowledgments

We would to thank the reviewers for their useful comments and suggestions, and Jingyi Zhang for providing the implementation of the phrase-based forced decoding for our experiments. This work was conducted under the program “Promotion of Global Communications Plan: Research, Development, and Social Demonstration of Multilingual Speech Translation Technology” of the Ministry of Internal Affairs and Communications (MIC), Japan.

References

Bangalore, S., Bordel, G., and Riccardi, G. (2001). Computing consensus translation from multiple machine translation systems. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 351–354.

Bentivogli, L., Bisazza, A., Cettolo, M., and Federico, M. (2016). Neural versus phrase-based machine translation quality: a case study. In *Proceedings of EMNLP*, pages 257–267, Austin, USA.

Cherry, C. and Foster, G. (2012). Batch tuning strategies for statistical machine translation. In *Proceedings of NAACL-HLT*, pages 427–436, Montréal, Canada.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of WMT*, pages 376–380, Baltimore, USA.

Devlin, J., Zbib, R., Huang, Z., Lamar, T., Schwartz, R., and Makhoul, J. (2014). Fast and robust neural network joint models for statistical machine translation. In *Proceedings of ACL*, pages 1370–1380, Baltimore, USA.

Dyer, C., Chahuneau, V., and Smith, N. A. (2013). A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of ACL*, pages 644–648, Atlanta, USA.

Ehling, N., Zens, R., and Ney, H. (2007). Minimum Bayes risk decoding for BLEU. In *Proceedings of ACL*, pages 101–104, Prague, Czech Republic.

Freitag, M., Huck, M., and Ney, H. (2014). Jane: Open source machine translation system combination. In *Proceedings of EACL*, pages 29–32, Gothenburg, Sweden.

Gimpel, K., Batra, D., Dyer, C., and Shakhnarovich, G. (2013). A systematic exploration of diversity in machine translation. In *Proceedings of EMNLP*, pages 1100–1111, Seattle, USA.

Goto, I., Chow, K. P., Lu, B., Sumita, E., and Tsou, B. K. (2013). Overview of the patent machine translation task at the NTCIR-10 workshop. In *Proceedings of the 10th NTCIR Conference on Evaluation of Information Access Technologies*, pages 260–286, Tokyo, Japan.

He, W., He, Z., Wu, H., and Wang, H. (2016). Improved neural machine translation with SMT features. In *Proceedings of AAAI*, pages 151–157.

Heafield, K. and Lavie, A. (2011). CMU system combination in WMT 2011. In *Proceedings of WMT*, pages 145–151, Edinburgh, Scotland.

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

Hildebrand, A. S. and Vogel, S. (2008). Combination of machine translation systems via hypothesis selection from combined n-best lists. In *Proceedings of AMTA*, pages 254–261.

Junczys-Dowmunt, M., Dwojak, T., and Sennrich, R. (2016). The AMU-UEDIN submission to the WMT16 news translation task: Attention-based NMT models as feature functions in phrase-based SMT. In *Proceedings of WMT*, pages 319–325, Berlin, Germany.

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 ACL*, pages 177–180, Prague, Czech Republic.

Koehn, P. and Knowles, R. (2017). Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver, Canada.

Le, H.-S., Allauzen, A., and Yvon, F. (2012). Continuous space translation models with neural networks. In *Proceedings of NAACL-HLT*, pages 39–48, Montréal, Canada.

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

Och, F. J. (2003). Minimum error rate training in statistical machine translation. In *Proceedings of ACL*, pages 160–167, Sapporo, Japan.

Och, F. J., Gildea, D., Khudanpur, S., Sarkar, A., Yamada, K., Fraser, A., Kumar, S., Shen, L., Smith, D., Eng, K., Jain, V., Jin, Z., and Radev, D. (2004). A smorgasbord of features for statistical machine translation. In Susan Dumais, D. M. and Roukos, S., editors, *Proceedings of HLT-NAACL*, pages 161–168, Boston, USA.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318, Philadelphia, USA.

Popović, M. (2017). chrF++: words helping character n-grams. In *Proceedings of WMT*, pages 612–618, Copenhagen, Denmark.

Sennrich, R., Birch, A., Currey, A., Germann, U., Haddow, B., Heafield, K., Miceli Barone, A. V., and Williams, P. (2017a). The University of Edinburgh's neural MT systems for WMT17. In *Proceedings of WMT*, pages 389–399, Copenhagen, Denmark.

Sennrich, R., Firat, O., Cho, K., Birch, A., Haddow, B., Hitschler, J., Junczys-Dowmunt, M., Läubli, S., Miceli Barone, A. V., Mokry, J., and Nadejde, M. (2017b). Nematus: a toolkit for neural machine translation. In *Proceedings of EACL*, pages 65–68, Valencia, Spain.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of ACL*, pages 1715–1725, Berlin, Germany.

Ueffing, N. and Ney, H. (2007). Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1):9–40.

Wang, Y., Cheng, S., Jiang, L., Yang, J., Chen, W., Li, M., Shi, L., Wang, Y., and Yang, H. (2017). Sogou neural machine translation systems for WMT17. In *Proceedings of WMT*, pages 410–415, Copenhagen, Denmark.

Zhang, J., Utiyama, M., Sumita, E., Neubig, G., and Nakamura, S. (2017). Improving neural machine translation through phrase-based forced decoding. In *Proceedings of IJCNLP*, pages 152–162, Taipei, Taiwan.

Zhou, L., Hu, W., Zhang, J., and Zong, C. (2017). Neural system combination for machine translation. In *Proceedings of ACL*, pages 378–384, Vancouver, Canada.