Statistic Machine Translation Boosted with Spurious Word Deletion

Shujie Liu[†], **Chi-Ho Li**[‡] and **Ming Zhou**[‡]
[†]School of Computer Science and Technology
Harbin Institute of Technology, Harbin, China
shujieliu@mtlab.hit.edu.cn
[‡]Microsoft Research Asia, Beijing, China
{chl, mingzhou}@microsoft.com

Abstract

Spurious words usually have no counterpart in other languages, and are therefore a headache in machine translation. In this paper, we propose a novel framework, skeleton-enhanced translation, in which a conventional SMT decoder can boost itself by considering the skeleton of the source input and the translation of such skeleton. By the skeleton of a sentence it is meant the sentence with its spurious words removed. We will introduce two models for identifying spurious words: one is a context-insensitive model, which removes all tokens of certain words; another is a context-sensitive model, which makes separate decision for each word token. We will also elaborate two methods to improve a translation decoder using skeleton translation: one is skeleton-enhanced re-ranking, which re-ranks the n-best output of a conventional SMT decoder with respect to a translated skeleton; another is skeleton-enhanced decoding, which re-ranks the translation hypotheses of not only the entire sentence but any span of the sentence. Our experiments show significant improvement (1.6 BLEU) over the state-of-the-art SMT performance.

1 Introduction

In language, words such as nouns, verbs, and conjunctions like "*because*", "*therefore*", etc. refer to objects, actions, events, or logical relationships. Their meaning is language-neutral and they usually have counterparts in another language. In contrast, some words serve to express (language-specific) grammatical relations only, and thus they may have no counterpart in another language. For the example[1] bilingual sentence in Figure 1, the Chinese words "*hen*" and "*bi*" have no counterparts on the other side, and so are the English words "*does*" and "*to*". We will call these words spurious words.

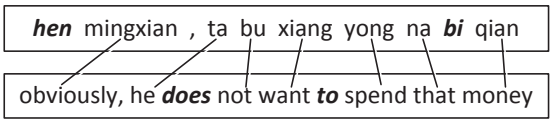

Figure 1. An example sentence pair with spurious words

To deal with the spurious words in sentence pairs, IBM models 3, 4 and 5 (Brown et al., 1993) introduce a special token *null*, which can align to a source/target word. Hence there are two types of words: spurious words (null-aligned) and non-spurious words. Similar with the IBM models, Fraser and Marcu (2007) proposed a new generative model called LEAF, in which words are classified into three types instead of two: spurious words, head words (which are the key words of a sentence) and non-head words (modifiers of head words).

These two methods are generative models for word alignment, which cannot be directly used in the conventional log-linear model of statistic machine translation (SMT). The conventional phrase-based SMT captures spurious words within the phrase pairs in the translation table. For example, in the phrase pair ("*yong* (*spend*) *na* (*that*) *bi qian*(*money*)", "*to spend that money*"), it implicitly deletes the source spurious word "*bi*" and implicitly inserts the target spurious word "*to*". The existence of spurious words in training data leads to certain kind of data sparseness. For example, "*na bi qian*" and "*na xie qian*" share the same translation ("*that money*"). If the spurious words ("*bi*" and

[1] For the examples in our paper, Chinese is the source language, and English is the target language.

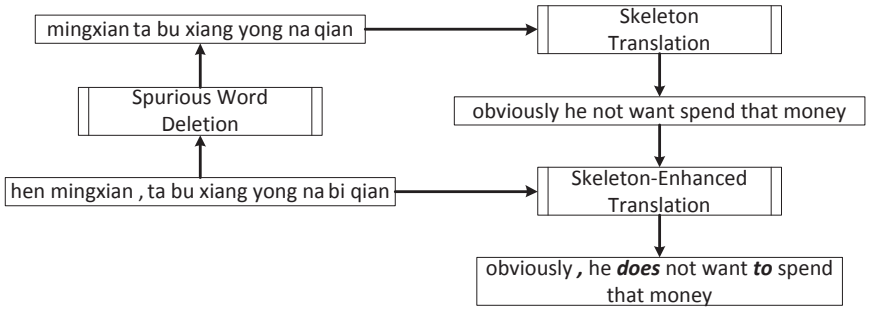

Figure 2. An example for skeleton-enhanced translation framework

"*xie*") are removed, then the two entries in translation table, and the associated statistics, can be combined into one. Moreover, because of the length constraint in phrase pairs, the existence of spurious words gets rid of many useful phrase pairs. For example, any phrase pair with the English side "*that money in the bank*" will not be recorded with length constraint smaller than five.

While spurious words lead to the harmful effect of data sparseness, they are useful in certain aspects in translation. For example, for the phrase pair ("*zhongguo*(*China*) *de yinhang*(*banks*)", "*the banks of China*"), the reordering of the two nouns is indicated by the spurious words "*de*"/"*of*".

In this paper we propose ***skeleton-enhanced translation***, which is a framework for balancing the merits and the noise of spurious words. We obtain the skeleton form of the source input sentence by removing all its spurious words. (In the example in Figure 2, the Chinese sentence "*hen mingxian , ta bu xiang yong na bi qian*" is converted into the skeleton "*mingxian ta bu xiang yong na qian*".) The source skeleton is then submitted to a ***skeleton translation*** system. (In the example, the source skeleton is translated into the target skeleton "*obviously he not want spend that money*".) Then we use the translated skeleton to help a conventional SMT decoder to select a better translation candidate. ("*obviously, he does not want to spend that money*" in the example.) That is, the translated skeleton is used in a softer and more adaptable way to improve the SMT performance. This approach maintains both the value of spurious words in reordering and the better generation capacity of skeleton phrase pairs.

Particularly, we introduce two skeleton-enhanced translation methods: one is ***skeleton-enhanced re-ranking***, which re-ranks the n-best output of a conventional SMT decoder with respect to translated skeleton; another is ***skeleton-enhanced decoding***, which re-ranks the translation hypotheses of not only the entire sentence but any span of the sentence.

In the following, we will elaborate the related works in Section 2, followed by the modules of skeleton-enhanced translation framework in Section 3, and the experiments results, which show significant improvements over a state-of-the-art phrase-based baseline system, in Section 4. Section 5 is our conclusion.

2 Related Work

Li et al. (2008) proposed three source spurious word deletion models[2] to calculate the translation probability for any source word to be translated into the special empty symbol, ε. The first model uses a uniform probability $p(\varepsilon)$, calculated by MLE from the word-aligned training corpus. The second one is word-type sensitive probability $p(\varepsilon|w)$, where w is the type of a source word, estimated in similar way as the first model. The third one is a word-token sensitive model, which uses Conditional Random Fields (CRF) (Lafferty et al., 2001) to calculate how likely a word token should be spurious given its context. All of them can improve a phrase-based system significantly.

Menezes and Quirk (2008) improved both phrase-based and treelet systems by introducing structural word insertion and deletion without requiring lexical anchor. The insertion/deletion order templates are based on syntactic cues and two additional feature functions (a count of structurally inserted words and a count of structurally deleted words). The probabilities of the templates are estimated by MLE.

[2] However, the problem of word insertion is discussed but not addressed.

In these two approaches, spurious words are treated the same as non-spurious words; they are simply assigned with a special translation probability or several special features. The data sparseness caused by the spurious words is still not solved.

3 Skeleton-Enhanced Translation

In this section we will elaborate two models for spurious word deletion (Section 3.1), translation of skeleton (Section 3.2), and two methods for skeleton-enhanced translation, viz. skeleton-enhanced re-ranking (Section 3.3) and skeleton-enhanced decoding (Section 3.4)

3.1 Spurious Word Deletion

Two methods for removing spurious words are investigated. One is a context-insensitive model (CIM), which removes all tokens of certain words; another is a context-sensitive model (CSM), which makes separate decision for each word token. Both methods use no more resources than conventional SMT translation modeling, viz. bilingual sentences with automatic word alignment. A word token is tagged as "spurious" if it is not aligned and "non-spurious" otherwise. The training instances of spurious word deletion for either the source language or the target language are thus obtained from word-aligned bilingual data.

CIM ranks the word types in order of decreasing percentage of spurious tokens among the alignment matrices of all training data. The top-n word types are then considered as spurious word types and all their tokens are removed from sentences.

CSM considers spurious word deletion as a sequential labeling problem, using CRF as the labeling algorithm. The context-sensitive features are similar to those in Li et al. (2008). The features for a word token w include:

1. The lexical form and the POS of the word w itself.
2. The lexical form and the POS of w_{-2}, w_{-1}, w_{+1}, w_{+2}, where w_{-2} and w_{-1} are the two words to the left of w, and w_{+1} and w_{+2} are the two words to the right of w.

The POS features, obtained from POS taggers, are of particular importance, as they do not only alleviate data sparseness but also help identify genuine spurious tokens. For example, the tokens of "*to*" as prepositions usually have counterparts in Chinese, whereas the tokens of "*to*" as infinitive-to do not.

As automatic word alignment is far from perfect, in order to keep a high precision of spurious word deletion, it is stipulated that a word token is not to be removed unless the model assigns a high probability to the deletion decision. Section 4.3 will explain how the probability threshold is selected so that precision and recall can be well balanced.

3.2 Skeleton Translation

After removal of spurious words from a source sentence, the resulting source skeleton is translated into the translated skeleton, which is a target language sentence with all target side spurious words removed. The translation of skeleton is also based on conventional SMT framework (including word alignment, phrase extraction, etc.). However, the training data and development data are all preprocessed with spurious word deletion as described in Section 3.1.

3.3 Skeleton-Enhanced Re-Ranking

In skeleton-enhanced re-ranking (SERR), the n-best output of a translation decoder, where the source input and the models have spurious words retained, are re-ranked in terms of the sentence-level BLEU score, using translated skeleton as reference. Following Watanabe et al. (2007), the approximated sentence-level BLEU score for a translation candidate $\check{e}$ of the source sentence $\check{f}$ is defined as the (document-level) BLEU score when $\check{e}$ is merged with the top one translation candidates of all source sentences other than $\check{f}$. The top four translated skeletons are taken as the references in calculating the sentence-level BLEU scores.

There is, however, a big problem in comparing the n-best translation candidates against the reference of translated skeletons. The n-best candidates still contain the target side spurious words whereas the translated skeletons have all the spurious words removed. Therefore we have to apply the target side spurious word deletion to the n-best candidates before the comparison. Of course, the re-ranked translation candidates will then have the target side spurious words recovered.

SERR can select better translation candidate with the better generation capacity of skeleton translation. For example, for the source sentence "*cong yin hang qu le yi da bi qian*", there are two

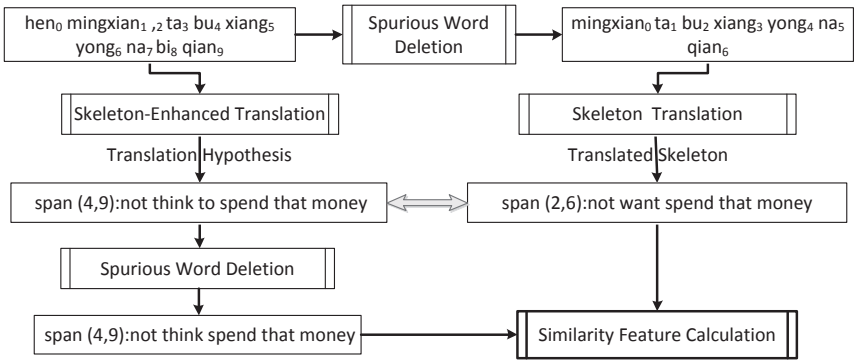

Figure 3. Calculation of similarity measures, used alongside the conventional features in SMT

candidates: "*fetch a lot of money from the bank*" vs. "*take a lot of money from the bank*". The source skeleton is "*yinhang qu qian*", and the translated skeleton is "*take money bank*". The translated skeleton selects a better translation, "*take*", for the source word "*qu*", since the skeleton language model gives much higher probability to "*take money bank*" than "*fetch money bank*". But the original language model will give similar probabilities to "*fetch a lot of money from the bank*" and "*take a lot of money from the bank*", since "*fetch*" is much farther from the words "*money*" and "*bank*".

3.4 Skeleton-Enhanced Decoding

The major drawback of SERR is that translated skeleton helps the selection/ranking of translation candidates only in the final step of the decoding process, viz. when the span under consideration ranges over the entire source input sentence. It would be much better if the selection/ranking of translation candidates in each span could be assisted by translated skeleton. That is what skeleton-enhanced decoding (SED) is about; the partial translation candidates are re-ranked with respect to the corresponding partial translated skeleton. There are two essential problems in SED:

1) Span mapping: how does SED know which span of the source/target skeleton corresponds to which span of a "source sentence"/"translation candidate"?
2) Span scoring: how does SED score partial translation candidates given the corresponding partial translated skeleton?

Regarding span mapping, when spurious words are removed from a source sentence, the position of a particular (non-spurious) word in the source sentence and its position in the source skeleton must be recorded. The mapping between these two kinds of position is thus established. Any span of a source sentence and its translation candidates, can be mapped to a particular span in the source skeleton, and can therefore also be mapped to the partial translated skeletons for this particular span in the source skeleton. For simplicity, only the top one partial translated skeleton in that span is used.

In the example shown in Figure 3, where the source sentence "*hen mingxian , ta bu xiang yong na bi qian*" has the skeleton "*mingxian ta bu xiang yong na qian*", the word position mapping enables us to map the source sentence span (4,9) "*bu xiang yong na bi qian*" to the source skeleton span (2,6) "*bu xiang yong na qian*". Then one of the translation candidates for the source sentence span (4,9) "*not think to spend that money*" can be mapped to the partial translated skeleton for the source skeleton span (2,6), e.g. "*not want spend that money*".

As to span scoring, the translation candidates of a source sentence span are scored with respect to their similarity with the corresponding partial translated skeleton. Note that, as in SERR, target language spurious word deletion must be applied to the translation candidates before calculating similarity. BLEU is no longer a suitable similarity measure here, since the comparison is between *incomplete* sentences. Instead we propose five features as similarity measure:

a) Unigram Precision (UP)
   The unigram precision feature is defined as $log(\frac{|T_S \cap S_T|}{|S_T|})$, where T_S stands for unigrams of the translated skeleton, and S_T stands for unigrams of the translation skeleton. $T_S \cap S_T$ stands for the intersection unigrams of T_S and S_T. For our example in

Figure 3, the feature value [3] should be *log(4/5)*.

b) Unigram Recall (UR)
The unigram recall feature is defined as $log(\frac{|T_S \cap S_T|}{|T_S|})$, and the feature value for our example in Figure 3 is *log(4/5)*.

c) Bigram Precision/Recall (BP/BR)
The Bigram Precision/Recall features are bigram versions of the Unigram Precision/Recall. The bigram precision feature value for our example in Figure 3 is *log(2/4)*, and the bigram recall feature value is also *log(2/4)*. All the bigram counts are stored to speed up the calculation, and new bigram will be generated by combining the border words when two spans are merged into a bigger one.

d) Skeleton Language Model (SLM)
This feature is the 4-gram language model score of the translation skeleton which is computed by the language model trained with spurious-word-deleted language model training data (in section 3.2).

The features of unigram/bigram recall/precision measure the similarity with respect to faithfulness while the feature of skeleton language model measures the similarity with respect to fluency[4]. The values of these five features are calculated on the fly during decoding. The five features are used alongside the conventional features in SMT, and the weights of all these features can be trained by any conventional method like Minimum Error Rate Training (MERT) (Och, 2003).

In sum, SED requires a mapping between (non-spurious) source word positions during source word deletion. During each decoding step, where the translation candidates of a particular source sentence span are under consideration, SED fetches the corresponding source skeleton span and its best partial translated skeleton, and then calculates several kinds of similarity between the (partial) translation candidates and the partial translated skeleton. The scoring of the translation candidates is thus enhanced by the skeleton-related features.

At a glance, SED is similar to collaborative decoding (Li et al., 2009, Liu et al., 2009). There is, however, a major difference. While every decoder involved in collaborative decoding selects its own best translation candidate by considering the candidates from other decoders, the SED decoder considers the candidates from skeleton translation, but not vice versa.

[3] Unigram precision is defined as -100 when there is no unigram in common between T_S and S_T

[4] Trigram and 4-gram features were attempted but found to give no further improvement.

4 Experiments

In this section, after elaborating the experiment settings in Section 4.1, we will explain how the thresholds for CIM and CSM are chosen empirically in Section 4.2 and Section 4.3 respectively. The improvement in SMT performance by the skeleton-enhanced methods will be shown in Section 4.4, followed by a detailed feature analysis for SED in Section 4.5.

4.1 Experiment Setting

We conduct our experiments on the test data from NIST 2006 and NIST 2008 Chinese-to-English machine translation tasks. To tune the model parameters, NIST 2005 test data is used as our development data. The bilingual training dataset is NIST 2008 training set excluding the Hong Kong Law and Hong Kong Hansard (contains 354k sentence pairs, 8M Chinese words and 10M English words). The translation pairs are extracted from word alignment matrices in the same way as Chiang (2007). The word alignment matrixes are generated in two directions by running GIZA++ (Och and Ney, 2003. Our 5-gram language model is trained from the Xinhua section of the Gigaword corpus.

There are two baseline SMT systems, one is a state-of-the-art implementation of hierarchical phrase-based model (Hiero) (Chiang, 2007) with conventional features; another is state-of-the-art implementation of Bracketing Transduction Grammar (Dekai Wu, 1997) (BTG) in CKY-style decoding with a lexical reordering model trained with maximum entropy (Xiong et al., 2006). The case-insensitive BLEU4 (Papineni et al., 2002) is used as the evaluation metric for SMT performance and statistical significance is performed using bootstrap re-sampling method proposed by Koehn (2004).

4.2 CIM of Spurious Words

For the context insensitive model (CIM) of spurious word deletion, we calculate the percentage of

unaligned tokens of each word type from word-aligned bilingual training corpus. The top 5 frequent unaligned words for source/target sentences are listed in Table 1.

Source		Target	
Word	Frequency	Word	Frequency
de	127,226	*the*	237,160
le	11,991	*of*	128,443
zai	6,577	*to*	41,217
zhong	6,250	*in*	39,146
shang	4,287	*for*	35,570

Table 1. Top 5 spurious words in Chinese/English

In order to confirm that skeleton translation can generate a better target skeleton than the conventional translation model, we compare the two sets of skeletons:

1) **Translation skeletons (TNS)**, which are baseline translation output with target side spurious words removed.
2) **Translated skeletons (TDS)**, which are output of skeleton translation.

If TDS with a set of spurious words outperforms the corresponding TNS a lot, the set of spurious words are good spurious words to build skeletons. The performance with different spurious words for CIM is shown in Table 2. In the table, the subscript to the label TDS/TNS indicates which spurious words are removed by the CIM. The numerals stand for the ranks of the removed spurious words (c.f. Table 1). The numerals to the left of dash are about source side spurious words and those to the right of dash are about target side spurious words.

From Table 2, we can find that two settings ($TDS_{1\text{-}1}$ and $TDS_{12\text{-}12}$) can get significant improvements on the translated skeleton compared with their corresponding baseline (the corresponding TSN: $TNS_{\text{-}1}$ and $TNS_{\text{-}12}$), which means, the skeleton translation using those spurious words can improve the performance of translated skeleton significantly. When more spurious words are included (for example, $TDS_{123\text{-}123}$ and $TDS_{1234\text{-}1234}$), the advantage of skeleton translation drops a lot, since too many non-spurious tokens may be removed using the arbitrary CIM model. In the flowing experiments of CIM, we will use these two settings $TDS_{1\text{-}1}$ and $TDS_{12\text{-}12}$ (called $CIM_{1\text{-}1}$ and $CIM_{12\text{-}12}$ in the following), which give the best performance. The phrase-table for CIM1-1 is reduced by 11% and that for CIM12-12 is 15%.

Skeletons	BLEU on Hiero	
	Nist'06	Nist'08
$TNS_{\text{-}1}$	32.71	25.54
$TDS_{1\text{-}1}$	**33.74(+1.03)**	**26.35(+0.81)**
$TNS_{\text{-}2}$	33.61	26.18
$TDS_{1\text{-}2}$	34.01(+0.4)	25.98(-0.2)
$TNS_{\text{-}1}$	32.71	25.54
$TDS_{2\text{-}1}$	**33.80(+1.09)**	25.77(+0.2)
$TNS_{\text{-}2}$	33.61	26.18
$TDS_{2\text{-}2}$	33.91(+0.3)	25.86(-0.3)
$TNS_{\text{-}12}$	31.91	24.80
$TDS_{12\text{-}12}$	**33.08(+1.17)**	**25.69(+0.89)**
$TNS_{\text{-}23}$	32.09	25.75
$TDS_{23\text{-}23}$	32.43(+0.34)	25.71(-0.04)
$TNS_{\text{-}123}$	31.41	24.39
$TDS_{123\text{-}123}$	31.90(+0.49)	24.15(-0.24)
$TNS_{\text{-}1234}$	30.82	24.04
$TDS_{1234\text{-}1234}$	31.15(+0.33)	23.73(-0.31)

Table 2. Performance of different CIMs

4.3 CSM of Spurious Words

Threshold	Precision		BLEU on Hiero	
	Source	Target	Nist'06	Nist'08
0.5↑	0.855	0.863	34.93	27.26
0.80↑	0.936	0.947	35.62	27.77
0.85↑	**0.962**	**0.976**	**36.16**	**28.41**
0.90↑	0.975	0.983	35.93	28.15
0.95↑	0.984	0.989	35.59	27.54

Table 3. Effect of probability thresholds in CSM

The context sensitive model (CSM) for spurious word deletion is trained with instances from word-aligned bilingual corpus and with the labeling system CRF++[5]. Note that the CRF tool assigns to each token a probability which means the likelihood the token is labeled as spurious. On the one hand, as show in the previous section, the removal of too many tokens is harmful to translation performance. On the other hand, the removal of too few tokens renders the skeleton-enhanced method useless. Thus we have to search for a good balance between precision and recall of spurious word deletion. This is achieved by setting a threshold on the probability of labeling as spurious word. That is, a token is not removed unless CSM assigns a probability value larger than certain threshold.

[5] http://crfpp.sourceforge.net/

System	BLEU on Hiero		BLEU on BTG	
	Nist'06	Nist'08	Nist'06	Nist'08
Baseline	34.49	26.95	33.26	25.53
$CIM_{1\text{-}1}$+SERR	35.02(+0.53)	27.30(+0.35)	33.67(+0.41)	25.69(+0.16)
$CIM_{1\text{-}1}$+SED	**35.43(+0.94)**	27.67(+0.72)	34.05(+0.79)	**26.35(+0.82)**
$CIM_{12\text{-}12}$+SERR	35.13(+0.64)	27.42(+0.47)	33.69(+0.43)	27.05(+0.52)
$CIM_{12\text{-}12}$+SED	**35.60(+1.11)**	**27.78(+0.83)**	**34.12(+0.92)**	**26.54(+1.01)**
CSM+SERR	**35.44(+0.85)**	**27.74(+0.79)**	33.99(+0.73)	**26.39(+0.86)**
CSM+SED	**36.16(+1.67)**	**28.41(+1.46)**	**34.62(+1.36)**	**27.02(+1.49)**

Table 4. Translation performance of different settings. Bold font indicates that the corresponding improvement in BLEU is statistically significant.

The precision of spurious word deletion and SMT performance with different thresholds (we remove the tokens whose deletion probabilities, given by the deletion model, are higher than the threshold, as said in Section 3.1) is shown in Table 3. With higher threshold, the deletion precisions for source and target increase monotonically. SMT performance on test sets increase significantly at first, then start dropping from the threshold 0.85. This shows that the value 0.85 is the optimal balance between the precision and recall of spurious word deletion, and thus this value will be adopted as the threshold for all subsequent experiments. There are 311,472 word tokens (of 853 types) deleted on the English side and 150,991 tokens (of 729 types) deleted on the Chinese side, and the corresponding phrase-table reduction is 17%.

4.4 Translation Results

In this section, both SERR and SED are evaluated against the two baselines Hiero and BTG. There are three different spurious word deletion models, which are $CIM_{1\text{-}1}$, $CIM_{12\text{-}12}$ and CSM (with threshold 0.85), and also two different skeleton-enhanced decoding methods: SERR and SED. For SERR, the size of the n-best list of translation candidates is 50.

In Table 4, all our skeleton-enhanced methods outperform the baseline system, which confirms the contribution of skeleton-enhanced translation. We can find that almost the similar results got for both Hiero and BTG compared with their baseline. The combination of $CIM_{12\text{-}12}$ and SED gets the highest SMT performance among all the CIM methods. SED methods are all better than SERR methods, which is reasonable, since SED methods re-rank not only the final n-best outputs, but also the partial translations. CSM method is better than CIM methods based on the same spurious-enhanced decoding model, since CIM method may arbitrarily delete tokens which have actual senses, but CSM can consider the context situation, which is more soft and sensitive. The combination of CSM and SED gets the highest performance among all the settings. It should be mentioned that $CIM_{12\text{-}12}$+SED, CSM+SERR and CSM+SED outperform the baseline system significantly according to the significant test proposed in Koehn (2004), with the two test data and two decoders, except only one result, which is CSM+SERR's result on Nist'06 with BTG decoder.

4.5 Feature Analysis for CSM+ SED

Setting	Features	BLEU on Hiero	
		Nist'06	Nist'08
X0	Baseline	34.49	26.95
X1	X0+SLM	35.01(+0.52)	27.42(+0.47)
X2	X0+UP	35.03(+0.54)	27.40(+0.45)
X3	X0+UR	**35.28(+0.79)**	**27.58(+0.63)**
X4	X0+BP	34.86(+0.37)	27.24(+0.39)
X5	X0+BR	35.02(+0.53)	27. 43(+0.48)

Table 5. Individual feature analysis

The individual contribution of each feature in SED (Section 3.4) is examined with the setting CSM+SED. The five features are skeleton language model (SLM), unigram precision (UP), unigram recall (UR), bigram precision (BP) and bigram recall (BR). As shown in Table 5, all the added features can improve the baseline system more or less. Among them, the most important feature is UR, which gives significant improvement (0.79) by itself on Nist'06. The significant improvement brought by UR means that the translated skeleton can help the skeleton-enhanced decoder to do a much better word selection. Preci-

sion features are less beneficial than recall features for both unigram and bigram. Compared with unigram and bigram features, SLM feature is of medium importance.

5 Conclusion

In this paper, we present a skeleton-enhanced translation framework, which contains spurious words deletion, skeleton translation, and skeleton-enhanced translation. The source sentence is converted, by spurious word deletion, into source skeleton, which is then translated into target skeleton. The translated skeleton is used as reference to assist the main decoder to improve the translation performance. Particularly, we introduced two spurious deletion models and two skeleton-enhanced translation methods. The two spurious deletion models are context insensitive methods to remove all the frequent spurious words in the sentence arbitrary, and a context sensitive deletion model using CRF. Our two skeleton-enhanced translation methods include a skeleton-enhanced re-ranking method using translated skeleton as reference skeletons, and a skeleton-enhanced decoding method which update all the partial translation results with the help of partial translated skeletons on the fly. Experiments show that our methods can improve the machine translation performance significantly. The context sensitive deletion method is much better than the context insensitive method, and the skeleton-enhanced decoding method is much better than the skeleton-enhanced re-ranking method.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Peitra, Robert L. Mercer. 1993. *The Mathematics of Statistical Machine Translation: Parameter Estimation*. *Computational Linguistics*, 19(2):263-311.

David Chiang. 2007. *Hierarchical Phrase-based Translation. Computational Linguistics*, 33(2).

Alexander Fraser and Daniel Marcu. 2007. *Getting the structure right for word alignment: LEAF*. In *Proceedings of* EMNLP, Pages: 51-60.

Philipp Koehn, Franz Josef Och and Daniel Marcu. 2003. *Statistical phrase-based translation*. In *Proceedings of* NAACL, Pages:48-54.

Philipp Koehn. 2004. *Statistical Significance Tests for Machine Translation Evaluation.* In *Proceedings of* EMNLP, Pages:388-395.

John Lafferty, Andrew McCallum and Fernando Pereira. 2001. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* In *Proceedings of* ICML, Pages:282-289.

Chi-Ho Li, Dongdong Zhang, Mu Li, Ming Zhou and Hailei Zhang. 2008. *An Empirical Study in Source Word Deletion for Phrase-based Statistical Machine Translation.* In *Proceedings of* the Third Workshop on Statistical Machine Translation, Pages:1-8.

Mu Li, Nan Duan, Dongdong Zhang, Chi-Ho Li, and Ming Zhou. 2009. *Collaborative decoding:partial hypothesis re-ranking using trnaslationconsen- sus between decoders.* In *Proceedings of* ACL, Pages: 585-592.

Yang Liu and Haitao Mi and Yang Feng and Qun Liu. 2009. *Joint Decoding with Multiple Translation Models.* In *Proceedings of* ACL, Pages: 576-584.

Arul Menezes and Chris Quirk. 2008. *Syntactic models for structural word insertion and deletion.* In *Proceedings of* EMNLP.

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

Franz Josef Och. 2003. *Minimum error rate training in statistical machine translation*. In *Proceedings of* ACL, Pages:160-167.

Franz Josef Och and Hermann Ney. 2003. *The Alignment Template Approach to Statistical Machine Translation. Computational Linguistics*, 30(4):417-449.

Kishore Papineni, Salim Roukos, Todd Ward and Weijing Zhu. 2002. *BLEU: a method for automatic evaluation of machine translation.* In *Proceedings of* ACL.

Taro Watanabe, Jun Suzuki, Hajinme Tsukada and Hideki Isozaki. 2007. *Online Large-Margin Training for Statistical Machine Translation*. In *Proceedings of* EMNLP. Pages:764-773.

Dekai Wu. 1997. *Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. Computational Linguistics*, 23(3).

Deyi Xiong, Qun Liu and Shouxun Lin. 2006. *Maximum entropy based phrase reordering model for statistical machine translation.* In *Proceedings of* ACL. Pages:521-528.

Philipp Koehn. 2004. *Statistical Significance Tests for Machine Translation Evaluation*. In *Proceedings of* EMNLP.