A Neural Network Classifier Based on Dependency Tree for English-Vietnamese Statistical Machine Translation

Viet Hong Tran[1,2], Quan Hoang Nguyen[2], and Vinh Van Nguyen[2]

[1]University of Economic and Technical Industries,Hanoi, Vietnam
[2]University of Engineering and Technology, Vietnam National University, Hanoi,Vietnam
thviet@uneti.edu.vn,quan94fm@gmail.com,vinhnv@vnu.edu.vn

Abstract Reordering in MT is a major challenge when translating between languages with different of sentence structures. In Phrase-based statistical machine translation (PBSMT) systems, syntactic pre-ordering is a commonly used pre-processing technique. This technique can be used to adjust the syntax of the source language to that of the target language by changing the word order of a source sentence prior to translation and solving to overcome a weakness of classical phrase-based translation systems: long distance reordering. In this paper, we propose a new pre-ordering approach by defining dependency-based features and using a neural network classifier for reordering the words in the source sentence into the same order in target sentence. Experiments on English-Vietnamese machine translation showed that our approach yielded a statistically significant improvement compared to our prior baseline phrase-based SMT system.

Key words: Natural Language Processing, Machine Translation, Phrase-based Statistical Machine Translation, Pre-ordering, Dependency Tree

1 Introduction

Recently the phrase-based and neural-based become dominant methods in current machine translation. Statistical machine translation (SMT) systems achieved a high performance in many typologically diverse language pairs. In phrase-based statistical machine translation (PBSMT) [1,2], syntactic pre-ordering is a commonly used pre-processing technique. It adjust the syntax of the source language to that of the target language by changing the word order of the source sentence prior to translation. This technology can overcome a weakness of classical phrase-based translation systems: long distance reordering. This is a major source of errors when translating between languages with difference of sentence structures. Phrase-based translation systems do not place a similar prior penalty on phrase reordering during decoding, however, such systems have been shown to profit from syntactic pre-ordering as well.

Many solutions to the reordering problem have been proposed, such as syntax-based model [3], lexicalized reordering [2], and tree-to-string methods [4]. Chiang [3] shows significant improvement by keeping the strengths of phrases, while incorporating syntax into SMT. Some approaches have been applied at the word-level [5]. They are particularly useful for language with rich morphology, for reducing data sparseness. Other kinds of syntax reordering methods require parser trees , such as the work in [6,5]. The parsed tree is more powerful in capturing the sentence structure. However, it is expensive to create tree structure, and building a good quality parser is also a hard task. All the above approaches require much decoding time, which is expensive.

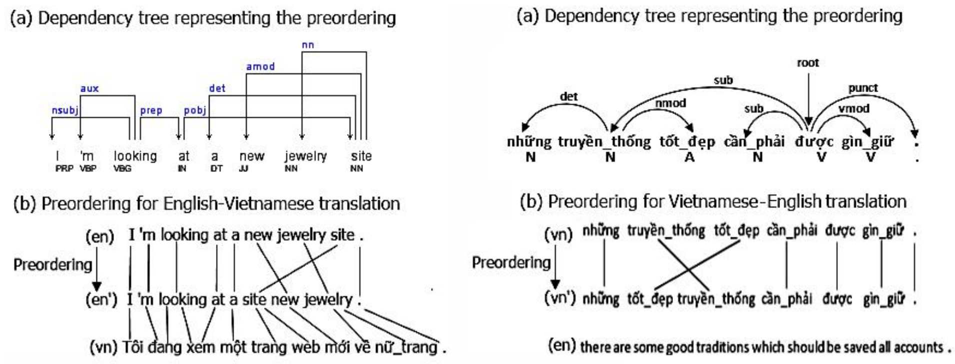


Figure 1. Example of preordering for English-Vietnamese translation and Vietnamese-English translation.

The end-to-end neural MT (NMT) approach [7] has recently been proposed for MT. The NMT system usually causes a serious out-of-vocabulary (OOV) problem, the translation quality would be badly affected ; The NMT decoder lacks a mechanism to guarantee that all the source words are translated and usually favors short translations. It is difficult for an NMT system to benefit from target language model trained on target monolingual corpus, which is proven to be useful for improving translation quality in statistical machine translation (SMT). NMT need much more training time. In [8], NMT requires longer time to train (18 days) compared to their best SMT system (3 days)

The approach we are interested in here is to balance the quality of translation with decoding time. Reordering approaches as a preprocessing step [9,10,11,12] are very effective (significant improvement over state of-the-art phrase-based and hierarchical machine translation systems and separately quality evaluation of each reordering models).

Inspired by this preprocessing approaches, we propose a combined approach which preserves the strength of phrase-based SMT in reordering and decoding time as well as the strength of integrating syntactic information in reordering. Firstly, the proposed method uses a dependency parsing for preprocessing step with training and testing. Secondly, transformation rules are applied to reorder the source sentences. The experimental resulting from English-Vietnamese pair shows that our approach achieved improvements in BLEU scores [13] compared to MOSES [14] which is the state of-the-art phrase-based SMT system.

This paper is structured as follows: Section 1 introduces the reordering problem, Section 2 reviews the related works. Section 3 briefly introduces classifier-based neural network Preordering for Phrase-based SMT. Section 4 describes experimental results. Section 5 discusses the experimental results. And, conclusions are given in Section 6.

2 Related works

The difference of the word order between source and target languages is the major problem in phrase-based statistical machine translation. Fig 1 describes an example that a reordering approach modifies the word order of an input sentence of a source languages (English) in order to generate the word order of a target languages (Vietnamese).

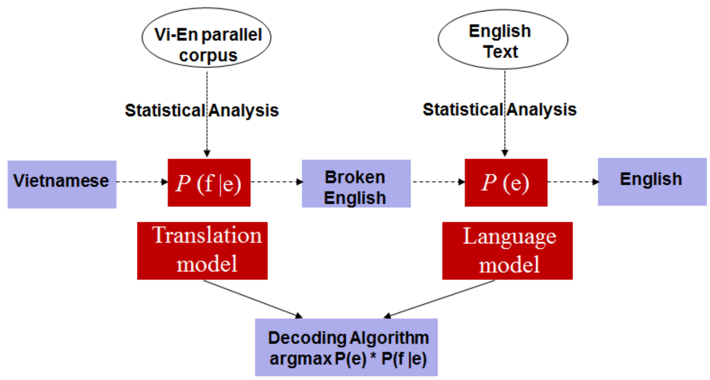


Figure 2. An example Phrase-based Statistical Machine Translation in Moses toolkit.

Many preordering methods using syntactic information have been proposed to solve the reordering problem. (Collin 2005; Xu 2009) [5,10] presented a preordering method which used manually created rules on parse trees. In addition, linguistic knowledge for a language pair is necessary to create such rules. Other preordering methods using automatic created reordering rules or a statistical classifier were studied [15,12]

Collins [5] developed a clause detection and used some handwritten rules to reorder words in the clause. Partly, (Habash 2007)[16] built an automatic extracted syntactic rules. Xu [10] described a method using a dependency parse tree and a flexible rule to perform the reordering of subject, object, etc... These rules were written by hand, but [10] showed that an automatic rule learner can be used.

Bach [17] propose a novel source-side dependency tree reordering model for statistical machine translation, in which subtree movements and constraints are represented as reordering events associated with the widely used lexicalized reordering models.

(Genzel 2010; Lerner and Petrov 2013) [11,12] described a method using discriminative classifiers to directly predict the final word order. Cai [18] introduced a novel pre-ordering approach based on dependency parsing for Chinese-English SMT.

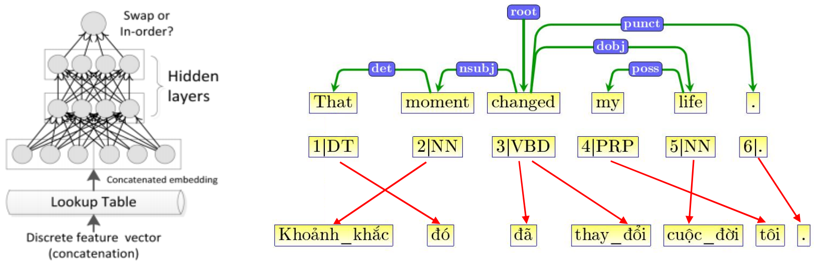
Isao Goto [19] described a preordering method using a target-language parser via cross-language syntactic projection for statistical machine translation.

Joachim Daiber [20] presented a novel examining the relationship between preordering and word order freedom in Machine Translation.

Chenchen Ding, [21] proposed extra-chunk pre-ordering of morphemes which allows Japanese functional morphemes to move across chunk boundaries.

Christian Hadiwinoto presented a novel reordering approach utilizing sparse features based on dependency word pairs [22] and presented a novel reordering approach utilizing a neural network and dependency-based embedding to predict whether the translations of two source words linked by a dependency relation should remain in the same order or should be swapped in the translated sentence [8]. This approach is complex and spend much time to process.

Our approach is closest similarity to [12], [8] but it has a few differences. Firstly, we aimed to develop the phrase-based translation model using dependency parse of source sentence to translate from English to Vietnamese. Secondly, we extracted automatically a set of English to Vietnamese transformation rules from English-Vietnamese parallel corpus by using Neural Network classification model with lexical and syntactic features based on dependency parsing of source sentence. Thirdly, we use the neural network



(a) (b)

Pair	Head			Left child			Right child			distance	Punctuation	Label
	x_h	$T(x_h)$	$L(x_h)$	x_c	$T(x_c)$	$L(x_c)$	x_c	$T(x_c)$	$L(x_c)$	$d(x_h,x_c)$	$w(x_h,x_c)$	
(3,2)	changed	VBD	root	moment	NN	nsubj	Null	Null	Null	-1	0	0
(3,5)	changed	VBD	root	Null	Null	Null	life	NN	dobj	+1	0	0
(2,1)	moment	NN	nsubj	That	DT	det	Null	Null	Null	-1	0	1
(5,4)	life	NN	dobj	my	PRP	poss	Null	Null	Null	-1	0	1

(c)

Pair	Left child				Right child				Head		Punctuation	Label
	x_l	$T(x_l)$	$L(x_l)$	$d(x_h, x_l)$	x_r	$T(x_r)$	$L(x_r)$	$d(x_h, x_r)$	x_h	$T(x_h)$	$w(x_l,x_r)$	
(2,5)	moment	NN	nsubj	-1	life	V	dobj	+1	changed	VBD	0	0

(d)

Figure 3. A Reordering Model for Statistical Machine Translation: (a) neural network classifier architecture; (b) an aligned English-Vietnamese parallel sentence pair with sample extracted training instances and features for (c) head-child classifier and (d) sibling classifier.

classifier to build two models that directly predict target-side word as a preprocessing step in phrase-based machine translation. As the same with [9,16], we also applied preprocessing in both training and decoding time.

3 A Neural Network Classifier-based Preordering for Phrase-based SMT

3.1 Phrase-based SMT

In this section, we will describe the phrase-based SMT system which was used for the experiments. Phrase-based SMT, as described by [1] translates a source sentence into a target sentence by decomposing the source sentence into a sequence of source phrases, which can be any contiguous sequences of words (or tokens treated as words) in the source sentence. For each source phrase, a target phrase translation is selected, and the target phrases are arranged in some order to produce the target sentence. A set of possible translation candidates created in this way were scored according to a weighted linear combination of feature values, and the highest scoring translation candidate was selected as the translation of the source sentence. Symbolically,

$$\hat{t} =_{t,a} \sum_{i=1}^{n} \lambda_i f_j(s,t,a) \tag{1}$$

when s is the input sentence, t is a possible output sentence, and a is a phrasal alignment that specifies how t is constructed from s, and $\hat{t}$ is the selected output sentence.

Feature	Description
Pair	Pair word with head-child relation
x_h	The head word x_h
$T(x_h)$	Part-of-speech (POS) tag of x_h
$L(x_h)$	The dependency label $L(x_h)$ linking x_h to head word of x_h
x_{cl}	The child word x_c if child left
$T(x_{cl})$	Part-of-speech (POS) tag of x_{cl}
$L(x_{cl})$	The dependency label L(xh) linking x_h to x_h
x_{cr}	The child word x_c if child right
$T(x_{cr})$	Part-of-speech (POS) tag of x_{cr}
$L(x_{cr})$	The dependency label $L(x_h)$ linking x_h to x_h
$d(x_h, x_c)$	The signed distance between the head and the child in the original source sentence: −2 if x_{cl} is on the left of x_h and there is at least one other child between them – 1 if x_{cl} is on the left of x_h and there is no other child between them +1 if x_{cr} is on the right of x_h and there is no other child between them + 2 if x_{cr} is on the right of x_h and there is no other child between them
$\omega\ (x_h, x_c)$	A Boolean $\omega(x_h, x_c)$ to indicate if any punctuation symbol, which is also the child of x_h, exists between x_h and x_c
Label	The label 1 or 0 indicates whether the two words need to be swapped or kept in order

(a) The feature of Head-child classifier

Feature	Description
Pair	Pair word with head-child relation
x_l	The left child word x_l
$T(x_l)$	Part-of-speech (POS) tag of x_l
$L(x_l)$	The dependency label $L(x_l)$ linking x_l to x_h
$d(x_h,x_l)$	the signed distance x_l to its head x_h +1 if x_{cr} is on the right of x_h and there is no other child between them + 2 if x_{cr} is on the right of x_h and there is no other child between them
x_r	The right child word x_r
$T(x_r)$	Part-of-speech (POS) tag of x_r
$L(x_r)$	The dependency label $L(x_r)$ linking x_r to x_h
$d(x_h,x_r)$	the signed distance x_r to its head x_h: −2 if x_{cl} is on the left of x_h and there is at least one other child between them – 1 if x_{cl} is on the left of x_h and there is no other child between them
x_h	The head word x_h
$T(x_h)$	Part-of-speech (POS) tag of x_h
$\omega(x_l, x_r)$	A Boolean $\omega(x_l, x_r)$ to indicate if any punctuation symbol, which is also the child of x_h, exists between x_l and x_r
Label	The label 1 or 0 indicates whether the two words need to be swapped or kept in order

(b) The feature of sibling classifier

Figure 4. (a) The feature of Head-child relation and (b) The feature of sibling relation used in training data from corpus English-Vietnamese

The weights λ_i associated with each feature f_i are tuned to maximize the quality of the translation hypothesis selected by the decoding procedure that computes the argmax. The log-linear model is a natural framework to integrate many features. The probabilities of source phrase given target phrases, and target phrases given source phrases, are estimated from the bilingual corpus.

[1] used the following distortion model (reordering model), which simply penalizes nonmonotonic phrase alignment based on the word distance of successively translated source phrases with an appropriate value for the parameter α:

$$d(a_i - b_{i-1}) = \alpha^{|a_i - b_{i-1} - 1|} \tag{2}$$

Current time, state-of-the-art phrase-based SMT system using the lexicalized reordering model in Moses toolkit. In our work, we also used Moses to evaluate on English-Vietnamese machine translation tasks. Fig 2 show an architecture of Phrase-based Statistical Machine Translation in Moses toolkit.

3.2 Classifier-based Preordering

In this section, we describe the learning model that can transform the word order of an input sentence to an order that is natural in the target language. English is used as source language, while Vietnamese is used as target language in our discussion about the word orders.

For example, when translating the English sentence:

That moment changed my life .

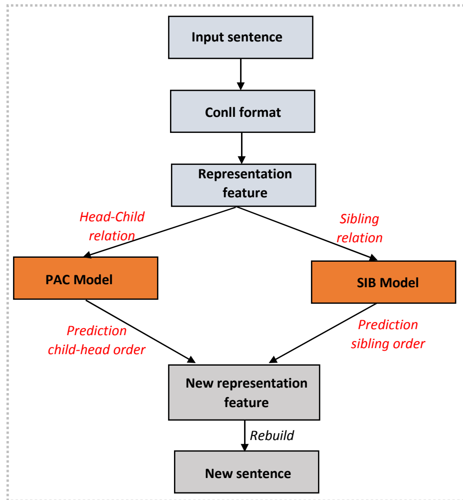


Figure 5. Framework for Preordering a new source sentence from parallel corpus.

to Vietnamese, we would like to reorder it as:

moment that changed life my .

And then, this model will be used in combination with translation model.

Training Data for Preordering and Features We use the dependency grammars and the differences of word order between English and Vietnamese to create a set of the reordering rules. With the POS tags and head-modifier dependencies shown in Figure 3, Traversing the dependency tree starting at the root to reordering. We determine the order of the head and its children for each head word and continue the traversal recursively in that order. In the above example, we need to decide the order of the head "changed" with the children "moment", "life"; the head "moment" with child "that", the head "life" with child "my".

The words in sentence are reordered by a new sequence learned from training data using two neural classifiers. The head-child classifier predicts the order of the translated words of a source word and its head word. The sibling classifier predicts the order of the translated words of two source words that both have the common head word.

The features extracted based on dependency tree and alignment information. We traverse the tree from the top, with each head-child and sibling relation we decide swap or no swap in dependency trees.

Classification Model We train two classifiers with a head-child relation and with a sibling relation. Each binary classifier takes a set of features related to the two source words as its input and predicts if the translated words should be swapped (positive) or remain in order (negative) each number of possible children. In hence, the classifiers learn to trade off between a rich set of overlapping features. List of features are given in Fig 4.

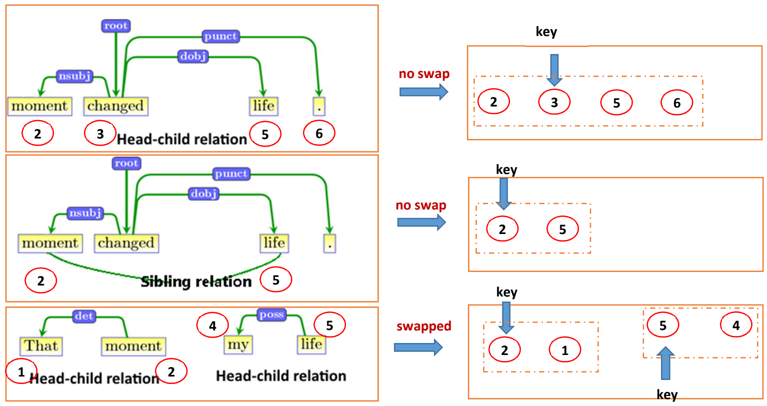


Figure 6. An Example for reordering after applying method classifier.

The classifier is a feed-forward neural network whose input layer contains the features. Each feature is mapped by a lookup table to a continuous vector representation. The resulting vectors are concatenated and fed into a series of hidden layers using the rectified linear activation function. Inspried from [8], we also initialize the hidden layers and the embedding layer for non-word features (POS tags, dependency labels, and Boolean indicators) by a random uniform distribution. For word features x_h, x_c, x_l, and x_r, we initialize their embeddings by the dependency-driven embedding scheme of (Bansal, Gimpel, and Livescu 2014) [23]. This scheme is a modified skip-gram model, which given an input word, predicts its context, resulting in a mapping such that words with similar surrounding words have similar continuous vector representations (Mikolov et al. 2013) [24].

The training instances for the neural network classifiers are obtained from a word-aligned parallel corpus with head-child or sibling relation are extracted from their corresponding order label, swapped or in order, depending on the positions of their aligned target-side words. The NN classifiers are trained using back-propagation to minimize the cross-entropy objective function.

The learning algorithm produces a sparse set of features. In our experiments the our models have typically only a few 130K non-zero feature weights English-Vietnamese language pairs.

When extracting the features, every word can be represented by its word identity, its POS-tags from the treebank, syntactic label. We also include pairs of these features, resulting in potentially bilexical features.

We describe a method to build training data for a pair English to Vietnamese. Our purpose is to reconstruct the word order of input sentence to an order that is arranged as Vietnamese words order. For example with the English sentence in Figure 3, after applying our framework in Fig 5 for prediction two relation (head-child relation, sibling relation) and reordering as described in Fig 6, the input sentence:

Algorithm 1 Build Models

```
input: dependency trees of source sentences
       and alignment pairs;
output: Two neural network classifier model:
       - PAC Model (Head-child relation Model)
       - SIB Model (Sibling relation Model)
for each head-child relation pair in dependency trees of subset
and alignment pairs of sentences do
   generate PAC_feature (head-child relation + label) ;
for each sibling relation pair in dependency trees of subset
and alignment pairs of sentences do
   generate SIB_feature (sibling relation + label) ;
end for
Build PAC model from set of PAC_features;
Build SIB model from set of PAC_features;
```

Algorithm 2 Reordering

```
input: a source sentence;
output: a new source sentence;
for each dependency tree of a source sentence do
   for each head-child relation in tree do
      prediction head-child order from PAC Model
   end for
   for each sibling relation in tree do
      prediction sibling order from SIB Model
   end for
end for
Build new sentence;
```

That moment changed my life .

is transformed into Vietnamese order:

moment that changed life my .

For this approach, we first do preprocessing to encode some special words and parser the sentences to dependency tree using Stanford Parser [25]. Then, we use target to source alignment and dependency tree to generate features. We add the information of the dependency tree as described in Fig 4 with each relation (head-child relation and sibling relation) from the dependency tree. For each family in the tree, we generate a training instance if it has less than and equal four children.

For every node in the dependency tree, from the top-down, we find the node matching against the pattern in classifier model, and if a match is found, the associated order applyed. We arrange the words in the English sentence, which is covered by the matching node, like Vietnamese words order. And then, we do the same for each children of this node.

The our algorithm's outline is given as Alg. 1 and Alg. 2

Algorithm 1 extract features and build models with input including dependency trees of source sentences and alignment pairs.

Algorithm 2 prediction order by considering head-child and sibling relation after finish Algorithm 1 from source-side dependency trees to build new sentence.

Corpus	Sentence pairs	Training Set	Development Set	Test Set
General	133403	131019	1304	1080

		English	Vietnamese
Training	Sentences	131019	
	Average Length	19.34	18.09
	Word	2534498	2370126
	Vocabulary	50118	56994
Development	Sentences	1304	
	Average Length	18.19	17.13
	Word	28773	27101
	Vocabulary	3713	3958
Test	Sentences	1080	
	Average Length	21.5	20.9
	Word	28036	27264
	Vocabulary	3918	4316

Table 1. Corpus Statistical

Name	Description
Baseline	Phrase-based system
Auto Rules	Phrase-based system with corpus which is be preprocessing using automatic rules
Our method	Phrase-based system with corpus which is be preprocessing using neural network Classifier

Table 2. Our experimental systems on English-Vietnamese parallel corpus

The reordering decisions are made by two classifiers (head-child classifier and sibling classifier) where class labels correspond to decide swapped or no swapped. We train a separate classifier for each relation. Crucially, we do not learn explicit tree transformations rules, but let the classifiers learn to trade off between a rich set of overlapping features. To build a classification model, we use neural network classification model in the Tensorflow tools [26].

We apply them in a dependency tree recursively starting from the root node. If the POS-tags of a node matches the left-hand-side of the rule, the rule is applied and the order of the sentence is changed. We go through all the children of the node and matching rules for them from the set of automatically rules.

Fig 5 gives framework of original and process phrase in English. After apply this framework, with the source sentence in English: " that moment changed my life .", and the target Vietnamese reordering " Khoảnh_khắc đó đã thay_đổi cuộc_đời tôi .". This sentences is arranged as the Vietnamese order. Vietnamese sentences are the output of our method. As you can see, after reordering, the original English has the same word order: "moment that changed life my ." in Figure 1.

4 Experiment

In this section, we present our experiments to translate from English to Vietnamese in a statistical machine translation system. The language pair chosen is English-Vietnamese. We used Stanford Parser [25] to parse source sentence (English sentences).

We used dependency parsing and rules extracted from training the features-rich discriminative classifiers for reordering source-side sentences. The rules are automatically extracted from English-Vietnamese parallel corpus and the dependency parser of English examples. Finally, they used these rules to reorder source sentences. We evaluated our approach on English-Vietnamese machine translation tasks with systems in table 2 which shows that it can outperform the baseline phrase-based SMT system.

We give some definitions for our experiments:

- Baseline: use the baseline phrase-based SMT system using the lexicalized reordering model in Moses toolkit.
- Auto Rules : the phrase-based SMT systems applying automatic rules.
- Our method: the Phrase-based system with corpus which is preprocessed using neural network Classifier.

4.1 Implementation

- We used Stanford Parser [25] to parse source sentence and apply to preprocessing source sentences (English sentences).
- We used neural network classifier in Tensorflow tools [26] for training the features-rich discriminative classifiers to build model and apply them for reordering words in English sentences according to Vietnamese word order.
- We implemented preprocessing step during both training and decoding time.
- Using the SMT Moses decoder [14] for decoding.
- Using Pre-trained word vector [27] and dependency-driven continous word representation [23] for the neural network classifiers.

4.2 Data set and Experimental Setup

We used an English-Vietnamese corpus [28], including about 131019 pairs for training, 1080 pairs for testing and 1304 pairs for development test set. Table 1 gives more statistical information about our corpora. We conducted some experiments with SMT Moses Decoder [14] and SRILM [29]. We trained a trigram language model using interpolate and kndiscount smoothing with Vietnamese mono corpus. Before extracting phrase table, we use GIZA++ [30] to build word alignment with grow-diag-final-and algorithm. Besides using preprocessing, we also used default reordering model in Moses Decoder: using word-based extraction (wbe), splitting type of reordering orientation to three classes (monotone, swap and discontinuous – msd), combining backward and forward direction (bidirectional) and modeling base on both source and target language (fe) [14]. To contrast, we tried preprocessing the source sentence with manual rules and automatically rules.

4.3 BLEU score

The result of experiments in table 3 show our method to process the source sentences. In this method, we can find out various phrases in the translation model. So that, they enable us to have more options for decoder to generate the best translation.

System	BLEU (%)
Baseline	26.51
Auto Rules	27.05
Our method	27.17

Table 3. Translation performance for the English-Vietnamese task

Table 3 describes the BLEU score of our experiments. As we can see, by applying preprocessing in both training and decoding, the BLEU score of our best system increase by 0.26 point over "Baseline system". Improvement over 0.26 BLEU point is valuable because baseline system is the strong phrase based SMT (integrating lexicalized reordering models). We also carried out the experiments with Automatic rules [31]. Using automatic rules help the phrased translation model generate some best translation. Besides, by applying two models to prediction right order in each relation: head-child relation and sibling relation, we propose a new preordering approach which there is no rules based in our framework. The result proved that the effect of applying our method on the dependency tree when the BLEU score is higher than baseline systems.

5 Analysis and Discussion

We have found that in our experiments work is sufficiently correlated to the translation quality done manually. Besides, we also have found some error causes such as parse tree source sentence quality, word alignment quality and quality of corpus. All the above errors can effect reordering in translation system.

We focus mainly on explore the rich dependency feature combine input representation with word-embedding to build two models: PAC model for head-child relation and SIB model for sibling relation based on neural network classifier. Our study employed dependency syntactic and applying these models to reorder the source sentence and applied to English to Vietnamese translation systems.

Based on these phenomena, translation quality has significantly improved. We carried out error analysis sentences and compared to the golden reordering. Our analysis has also the benefits of our method on translation quality. In combination with machine learning method in related work [12], it is shown that applying classifier method to solve reordering problems automatically.

6 Conclusion

In this study, we propose a new pre-ordering approach for English-Vietnamese Statistical Machine Translation by defining dependency-based features and using a neural network classifier for reordering the words in the source sentence into the same order in target sentence. We used a neural network classifier in Tensorflow for training the features-rich discriminative classifiers and reordering words in English sentence according to Vietnamese word order.

We evaluated our approach on English-Vietnamese machine translation tasks. Experiments on English-Vietnamese machine translation show that our approach yields a statistically significant improvement compared to our prior baseline phrase-based SMT system. The experimental results showed that our approach achieved statistical improvements over a state-of-the-art phrase-based baseline system by BLEU point scores. We believe that such reordering rules benefit English-Vietnamese language pairs.

In the future, we plan to further investigate in this direction and use our method on other language pairs. We also attempt to create more efficient preordering rules by exploiting the rich information in dependency structures.

7 Acknowledgements

This work described in this paper has been partially funded by TC.02-2016-03 project: Building a machine translation system to support translation of documents between Vietnamese and Japanese to help managers and businesses in Hanoi approach Japanese market.

References

1. Koehn, P., Och, F.J., Marcu, D.: Statistical phrase-based translation. In: Proceedings of HLT-NAACL 2003, Edmonton, Canada (2003) 127–133
2. Och, F.J., Ney, H.: The alignment template approach to statistical machine translation. Computational Linguistics **30**(4) (2004) 417–449
3. Chiang, D.: A hierarchical phrase-based model for statistical machine translation. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), Ann Arbor, Michigan (June 2005) 263–270
4. Zhang, Y., Zens, R., Ney, H.: Chunk-level reordering of source language sentences with automatically learned rules for statistical machine translation. In: Proceedings of SSST, NAACL-HLT 2007 / AMTA Workshop on Syntax and Structure in Statistical Translation. (2007) 1–8
5. Collins, M., Koehn, P., Kucerová, I.: Clause restructuring for statistical machine translation. In: Proc. ACL 2005, Ann Arbor, USA (2005) 531–540
6. Quirk, C., Menezes, A., Cherry, C.: Dependency treelet translation: Syntactically informed phrasal smt. In: Proceedings of ACL 2005, Ann Arbor, Michigan, USA (2005) 271–279
7. Wu, Y., Schuster, M., Chen, Z., Le, Q.V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., et al.: Googles neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144, 2016
8. Hadiwinoto, C., Ng, H.T.: A dependency-based neural reordering model for statistical machine translation. arXiv preprint arXiv:1702.04510, 2017
9. Xia, F., McCord, M.: Improving a statistical mt system with automatically learned rewrite patterns. In: Proceedings of Coling 2004, Geneva, Switzerland, COLING (Aug 23–Aug 27 2004) 508–514
10. Xu, P., Kang, J., Ringgaard, M., Och, F.: Using a dependency parser to improve smt for subject-object-verb languages. In: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Boulder, Colorado, Association for Computational Linguistics (June 2009) 245–253
11. Genzel, D.: Automatically learning source-side reordering rules for large scale machine translation. In: Proceedings of the 23rd International Conference on Computational Linguistics. COLING '10, Stroudsburg, PA, USA, Association for Computational Linguistics (2010) 376–384
12. Lerner, U., Petrov, S.: Source-side classifier preordering for machine translation. In: EMNLP. (2013) 513–523
13. Papineni, Kishore, S.R.T.W., Zhu, W.: Bleu: A method for automatic evaluation of machine translation. In: ACL. (2002)

14. 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., Herbst, E.: Moses: Open source toolkit for statistical machine translation. In: Proceedings of ACL, Demonstration Session. (2007)
15. Yang, N., Li, M., Zhang, D., Yu, N.: A ranking-based approach to word reordering for statistical machine translation. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, Association for Computational Linguistics (2012) 912–920
16. Habash, N.: Syntactic preprocessing for statistical machine translation. Proceedings of the 11th MT Summit (2007)
17. Bach, N., Gao, Q., Vogel, S.: Source-side dependency tree reordering models with subtree movements and constraints. In: Proceedings of the Twelfth Machine Translation Summit (MTSummit-XII), Ottawa, Canada, International Association for Machine Translation (August 2009)
18. Jingsheng Cai, Masao Utiyama, E.S.Y.Z.: Dependency-based pre-ordering for chinese-english machine translation. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. (2014)
19. Goto, I., Utiyama, M., Sumita, E., Kurohashi, S.: Preordering using a target-language parser via cross-language syntactic projection for statistical machine translation. ACM Transactions on Asian and Low-Resource Language Information Processing **14**(3) (2015) 13
20. Daiber, J., Stanojevic, M., Aziz, W., Sima'an, K.: Examining the relationship between preordering and word order freedom in machine translation. In: Proceedings of the First Conference on Machine Translation (WMT16), Berlin, Germany, August. Association for Computational Linguistics. (2016)
21. Ding, C., Sakanushi, K., Touji, H., Yamamoto, M.: Inter-, intra-, and extra-chunk preordering for statistical japanese-to-english machine translation. ACM Trans. Asian Low-Resour. Lang. Inf. Process. **15**(3) (January 2016) 20:1–20:28
22. Hadiwinoto, C., Liu, Y., Ng, H.T.: To swap or not to swap? exploiting dependency word pairs for reordering in statistical machine translation. In: Thirtieth AAAI Conference on Artificial Intelligence. (2016)
23. Bansal, M., Gimpel, K., Livescu, K.: Tailoring continuous word representations for dependency parsing. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). (June 2014) 809–815
24. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. International Conference on Learning Representations (ICLR) Workshop (2013)
25. Cer, D., de Marneffe, M.C., Jurafsky, D., Manning, C.D.: Parsing to stanford dependencies: Trade-offs between speed and accuracy. In: 7th International Conference on Language Resources and Evaluation (LREC 2010). (2010)
26. : TensorFlow: Large-scale machine learning on heterogeneous systems. (2015) Software available from tensorflow.org.
27. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv preprint arXiv:1607.04606 (2016)
28. Nguyen, T.P., Shimazu, A., Ho, T.B., Nguyen, M.L., Nguyen, V.V.: A tree-to-string phrase-based model for statistical machine translation. In: Proceedings of the Twelfth Conference on Computational Natural Language Learning (CoNLL 2008), Manchester, England, Coling 2008 Organizing Committee (August 2008) 143–150
29. Stolcke, A.: Srilm - an extensible language modeling toolkit. In: Proceedings of International Conference on Spoken Language Processing. Volume 29. (2002) 901–904
30. Och, F.J., Ney, H.: A systematic comparison of various statistical alignment models. Computational Linguistics **29**(1) (2003) 19–51
31. Tran, V.H., Vu, H.T., Nguyen, V.V., Nguyen, M.L.: A classifier-based preordering approach for english-vietnamese statistical machine translation. 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2016)